Personal Computer

PC 10/20/40
A 2000

MS-DOS 3.2
Referenzhandbuch

Microsoft® MS®-DOS

Commodore

Inhaltsverzeichnis

1. Einführung 1– 1
1.1 Was ist MS–DOS? 1– 1
1.2 Über dieses Handbuch 1– 1
1.3 Die MS–DOS–Dateien 1– 2

2. System–Konfigurierung 2– 1
2.1 Was ist eine Konfigurationsdatei? 2– 1
2.2 Die CONFIG.SYS–Datei erzeugen 2– 1
2.3 Die CONFIG.SYS–Befehle im einzelnen 2– 1

BREAK 2– 4
BUFFERS 2– 5
COUNTRY 2– 6
DEVICE 2– 7
DRIVPARM 2– 8
FCBS 2– 10
FILES 2– 11
LASTDRIVE 2– 12
SHELL 2– 13

2.4 Ein Beispiel 2– 14
2.5 Installierbare Gerätetreiber 2– 14
2.5.1 ANSI.SYS 2– 15
2.5.1.1 Cursor–Steuerfunktionen 2– 16
2.5.1.2 Lösch–Funktionen 2– 18
2.5.1.3 Grafische Funktionen 2– 18
2.5.1.4 Umdefinition der Tastaturbelegung 2– 21
2.5.2 DRIVER.SYS 2– 22
2.5.3 RAMDRIVE.SYS 2– 23

3. Festplatten–Konfigurierung mit FDISK 3– 1
3.1 Einleitung 3– 1
3.2 FDISK starten 3– 2
3.2.1 Eine DOS–Partition einrichten 3– 3
3.2.2 Die aktuelle Partition ändern 3– 5
3.2.3 Eine DOS–Partition löschen 3– 6
3.2.4 Partitions–Informationen anzeigen 3– 6
3.2.5 Eine weitere Festplatte partitionieren 3– 7

4. Dateien und Verzeichnisse 4– 1
4.1 Dateien 4– 1
4.1.1 Was ist eine Datei? 4– 1
4.1.2 Wie MS–DOS Dateien verwaltet 4– 1
4.1.3 Ein Verzeichnis anzeigen: Der DIR–Befehl 4– 2
4.1.4 Eine Disk überprüfen: Der CHKDSK–Befehl 4– 4
4.1.5 Wie Dateien benannt werden 4– 5
4.1.6 Verbotene Dateinamen 4– 6
4.1.7 Jokerzeichen für globale Dateinamen 4– 7
4.1.7.1 Das ?–Jokerzeichen 4– 7
4.1.7.2 Das *–Jokerzeichen 4– 7
4.1.8 Wie Dateien geschützt werden können 4– 8
4.2 Disk–Verzeichnisse 4– 8
4.2.1 Verzeichnis–Pfade 4– 11
4.2.1.1 Pfad–Namen 4– 12
4.2.1.2 Pfade und externe Befehle 4– 13
4.2.1.3 Pfade und interne Befehle 4– 14
4.2.2 Die Arbeit mit Verzeichnissen 4– 15
4.2.2.1 Das aktuelle Verzeichnis anzeigen 4– 15
4.2.2.2 Ein Verzeichnis anlegen 4– 16
4.2.2.3 Das aktuelle Verzeichnis wechseln 4– 17
4.2.2.4 Ein Verzeichnis löschen 4– 17
4.2.2.5 Ein Verzeichnis umbenennen 4– 18

5. Allgemeines über Befehle 5– 1
5.1 Was ist ein Befehl? 5– 1
5.2 Befehlsarten bei MS–DOS 5– 1
5.3 Befehlsparameter 5– 2
5.4 Was Sie bei der Befehlseingabe wissen sollten 5– 4
5.5 Ein– und Ausgabe 5– 5
5.5.1 Umleiten der Standard–Ein– und –Ausgabe 5– 5
5.5.2 Filter 5– 6
5.5.2.1 Befehlsverkettung mit Filtern 5– 7
5.6 Stapelverarbeitung 5– 8
5.6.1 Die AUTOEXEC.BAT–Datei 5– 10
5.6.2 Wie eine AUTOEXEC.BAT–Datei erzeugt wird 5– 11
5.6.3 Eine Stapeldatei mit austauschbaren Parametern erzeugen 5– 12
5.6.4 Eine Stapeldatei abarbeiten 5– 13
5.7 Zusammenfassung 5– 14

6. Die MS–DOS–Befehle 6– 1
6.1 Übersicht 6– 1
6.2 Allgemeines Format der Befehlsbeschreibungen 6– 2
6.3 Die MS–DOS–Befehle im einzelnen 6– 3

APPEND 6– 6
ASSIGN 6– 8
ATTRIB 6– 10
BACKUP 6– 12
BREAK 6– 16
CHDIR 6– 17
CHKDSK 6– 19
CLS 6– 22
COMMAND 6– 23
COPY 6– 25
CTTY 6– 32
DATE 6– 33
DEL 6– 35
DIR 6– 37
DISKCOMP 6– 40
DISKCOPY 6– 43
ERASE 6– 46
EXE2BIN 6– 47
EXIT 6– 50
FDISK 6– 51
FIND 6– 52
FORMAT 6– 54
GRAFTABL 6– 59
GRAPHICS 6– 60
JOIN 6– 62
KEYBxx 6– 66
LABEL 6– 68
MKDIR 6– 70
MODE 6– 71
MORE 6– 76
PATH 6– 77
PRINT 6– 79
PROMPT 6– 84
RECOVER 6– 86
REN 6– 88
REPLACE 6– 90
RESTORE 6– 93
RMDIR 6– 97

SET 6– 98
SHARE 6–100
SORT 6–101
SUBST 6–103
SYS 6–106
TIME 6–107
TREE 6–109
TYPE 6–110
VER 6–112
VERIFY 6–113
VOL 6–114
XCOPY 6–115

6.3.2 Stapelverarbeitungsbefehle 6–119

ECHO 6–120
FOR 6–121
GOTO 6–123
IF 6–124
PAUSE 6–126
REM 6–128
SHIFT 6–128

7. MS–DOS–Edier– und Funktionstasten 7– 1
7.1 Spezielle MS–DOS–Edier– und Bedienungstasten 7– 1
7.2 Funktionstasten 7– 3
7.3 Beispiele 7– 4

8. Der Zeileneditor EDLIN 8– 1
8.1 Einführung 8– 1
8.2 EDLIN starten 8– 1
8.3 EDLIN beenden 8– 3
8.4 Spezielle Edier–Tasten 8– 3
8.5 Die EDLIN–Befehle: Eine Übersicht 8– 15
8.6 Hinweise zur Benutzung der EDLIN–Befehle 8– 15
8.7 Parameter zur Befehlserweiterung 8– 17
8.8 Die EDLIN–Befehle im einzelnen 8– 19

Append 8– 20
Copy 8– 21
Delete 8– 22
Edit 8– 25
End 8– 27

Insert . . . . . 8 – 28
List . . . . . 8 – 32
Move . . . . . 8 – 34
Page . . . . . 8 – 36
Quit . . . . . 8 – 37
Replace . . . . . 8 – 38
Search . . . . . 8 – 41
Transfer . . . . . 8 – 44
Write . . . . . 8 – 45

9. Dateivergleich mit FC . . . . . 9 – 1
9.1 Einführung . . . . . 9 – 1
9.2 Einschränkungen beim Vergleich . . . . . 9 – 1
9.3 Der Aufruf von FC . . . . . 9 – 2
9.4 Die FC–Schalter . . . . . 9 – 3
9.5 Wie Dateiunterschiede angezeigt werden . . . . . 9 – 6
9.6 Umleitung der FC–Ausgabe in eine Datei . . . . . 9 – 6
9.7 Beispiele . . . . . 9 – 7

10. LINK: Ein Bindeprogramm für Objekt–Dateien . . . . . 10 – 1
10.1 Einführung . . . . . 10 – 1
10.2 LINK aufrufen und benutzen . . . . . 10 – 2
10.2.1 Parameteranforderung . . . . . 10 – 2
10.2.2 Der vollständige Aufruf . . . . . 10 – 5
10.2.3 Angabe einer Parameterdatei . . . . . 10 – 7
10.2.4 Angabe von Suchpfaden für Bibliotheken . . . . . 10 – 9
10.2.5 Die List–Datei (MAP–Datei) . . . . . 10 – 11
10.2.6 Die temporäre Arbeitsdatei VM.TMP . . . . . 10 – 13
10.3 Befehlserweiterungen mit den LINK–Optionen . . . . . 10 – 13
10.3.1 Die LINK–Optionen im einzelnen . . . . . 10 – 14

/CPARMAXALLOC . . . . . 10 – 15
/DOSSEG . . . . . 10 – 17
/DSALLOCATE . . . . . 10 – 18
/EXEPACK . . . . . 10 – 19
/HELP . . . . . 10 – 20
/HIGH . . . . . 10 – 21
/LINENUMBERS . . . . . 10 – 22
/MAP . . . . . 10 – 23
/NODEFAULTLIBRARYSEARCH . . . . . 10 – 24
/NOGROUPASSICIATION . . . . . 10 – 25
/NOIGNORECASE . . . . . 10 – 26
/OVERLAYINTERRUPT . . . . . 10 – 27

/PAUSE 10–28
/SEGMENTS 10–30
/STACK 10–31

10.4 Wie LINK arbeitet 10–32
10.4.1 Segment–Ausrichtung 10–32
10.4.2 Rahmenzahl 10–33
10.4.3 Segment–Reihenfolge 10–33
10.4.4 Kombinierte Segmente 10–33
10.4.5 Gruppen 10–35
10.4.6 Fixierung aller nichtzugeordneten Referenzen 10–35

11. Das Testhilfeprogramm DEBUG 11– 1
11.1 Einführung 11– 1
11.2 DEBUG starten 11– 1
11.3 Befehlsstruktur 11– 3
11.4 Befehlsparameter 11– 4
11.5 Die DEBUG–Befehle im einzelnen 11– 7

Assemble 11– 8
Compare 11–11
Dump 11–12
Enter 11–14
Fill 11–16
Go 11–17
Hex 11–19
Input 11–20
Load 11–21
Move 11–23
Name 11–24
Output 11–27
Quit 11–28
Register 11–29
Search 11–32
Trace 11–33
Unassemble 11–35
Write 11–37

11.6 Fehlermeldungen bei DEBUG 11–39

Anhang A: Meldungen und Fehlermeldungen A–1
A.1 Disk– und Gerätefehlermeldungen A–1
A.2 Alphabetisches Verzeichnis aller MS–DOS–Meldungen A–4

1. Einführung

1.1 Was ist MS–DOS?

MS–DOS von Microsoft ist ein **Betriebsystem**. Ein Betriebssystem ist grundsätzlich die Zusammenstellung einer Reihe von Computer–Programmen, die es Ihnen erlauben, den Computer zu bedienen und für sich arbeiten zu lassen. MS–DOS ist sozusagen Ihr "stiller Partner", wenn Sie mit Ihrem Computer arbeiten. Sie können es mit der Elektrizitätsversorgung in einem Wohnhaus vergleichen: Sie benötigen diese zum Betrieb z.B. des Kühlschrankes oder der Waschmaschine, aber werden meistens ihrer Gegenwart gar nicht gewahr.

MS–DOS ist ein **Disk**–Betriebssystem. Das bedeutet, daß die einzelnen Programme von einer Diskette oder einer Festplatte geladen und benutzt werden können. Eine Festplatte hat eine wesentlich höhere Speicherkapazität, als eine Diskette, kann also wesentlich mehr Informationen speichern.

Nachdem Sie MS–DOS in den Speicher Ihres Computers geladen haben, können Sie Computer–Spiele spielen, Programme ablaufen lassen, Briefe oder andere Texte entwerfen oder auch mit einer Programmiersprache wie GW BASIC eigene Programme entwickeln.

Mit MS–DOS können Sie außerdem Geräte, wie Drucker oder Disk–Laufwerke an Ihrem Computer betreiben.

Im weiteren Verlauf dieses Handbuches wird wiederholt der Begriff **Disk** verwendet. In diesem Fall sind sowohl Disketten als auch Festplatten gemeint, da beide in Bezug auf die Datenspeicherung von MS–DOS gleich organisiert werden.

1.2 Wie Sie dieses Handbuch benutzen

Dieses Handbuch ist ein Referenzhandbuch für MS–DOS. Ein weiteres Handbuch, das ***MS–DOS–Benutzerhandbuch***, sollten Sie gelesen haben, ehe Sie mit diesem Handbuch arbeiten. Sie sollten also wissen, wie MS–DOS gestartet wird, wie Dateien kopiert, gelöscht oder umbenannt werden, wie man Disketten dupliziert und wie Anwenderprogramme gestartet werden. Sie sollten bereits Duplikate der Systemdisketten angefertigt, und diese an einem sichernen Ort verwart haben.

Die folgende Aufstellung soll Ihnen helfen, schnell die gewünschte Information in diesem Handbuch zu finden:

Wenn Sie wissen wollen. . . .	**suchen Sie in Kapitel**
wie das System konfiguriert wird	2
was Gerätetreiber sind	2
wie eine Festplatte vorbereitet wird	3
wie man Dateien benennt	4
was hierarchische Diskverzeichnisse sind	4
was Pfade sind	4
was ein Befehl ist	5
wie eine Stapeldatei erstellt wird	5
was ein bestimmter Befehl bewirkt	5
wie die MS–DOS–Ediertasten funktionieren	7
wie der Zeileneditor EDLIN benutzt wird	8
wie der Dateivergleich mit FC funktioniert	9
wie der Binder LINK benutzt wird	10
wie man Programmfehler mit DEBUG sucht	11
was eine Fehlermeldung bedeutet	Anhang A

1.3 Die MS–DOS–Dateien

Die beiden MS–DOS–Systemdisketten enthalten die folgenden Dateien:

Datei	**Funktion**
ANSI.SYS	definiert die ANSI–ESCAPE–Sequenzen
APPEND.COM	setzt einen Suchpfad für Datendateien
ASSIGN.COM	weist einem anderen Laufwerk eine Laufwerksbezeichnung zu
ATTRIB.EXE	setzt oder löscht Dateiattribute für die Zugriffsart
BACKUP.EXE	erstellt Sicherungskopien von Dateien
CHKDSK.EXE	überprüft Disks
COMMAND.COM	MS–DOS–Befehlsprozessor
DEBUG.EXE	erlaubt die Fehlersuche in Programmen
DISKCOMP.EXE	vergleicht Diskinhalte
DISKCOPY.EXE	erstellt Disketten–Duplikate
DRIVER.SYS	konfiguriert installierbare Block–Gerätetreiber
EDLIN.EXE	Zeileneditor
EXE2BIN.EXE	wandelt .EXE–Dateien in .COM–Dateien um

FC.EXE	vergleicht Dateien
FDISK.EXE	konfiguriert die Festplatte für MS–DOS
FIND.EXE	durchsucht Dateien nach einer Zeichenkette
FORMAT.EXE	formatiert Disks
GRAFTABL.EXE	lädt eine Tabelle mit Grafikzeichen
GRAPHICS.EXE	bereitet die Druckausgabe von Grafiken vor
GWBASIC.EXE	der GWBASIC–Interpreter
*IO.SYS	Betriebssystemschnittstelle zur Hardware
JOIN.EXE	verkettet eine Laufwerksbezeichnung mit einem angegebenen Pfadnamen
KEYBFR	Tastaturprogramm französisch
KEYBGR	Tastaturprogramm deutsch
KEYBIT	Tastaturprogramm italienisch
KEYBSP	Tastaturprogramm spanisch
KEYBUK	Tastaturprogramm englisch
LABEL.EXE	benennt eine Disk neu oder um
LINK.EXE	der Binder für Objektdateien
MODE.EXE	setzt Ausgabe– und Anzeigemodi
MORE.COM	erlaubt die bildschirmweise Anzeige von Daten
*MSDOS.SYS	MS–DOS–Betriebssystem–Kern
PRINT.EXE	druckt Dateien im Hintergrund
RAMDRIVE.SYS	installiert eine virtuelle Disk im Hauptspeicher
RECOVER.EXE	stellt Dateien mit fehlerhaften Sektoren wieder her
REPLACE.EXE	ersetzt ältere Dateiversionen durch neuere
RESTORE.EXE	restauriert Dateien aus Sicherungskopien
SHARE.EXE	istalliert die Unterstützung für gemeinsamen Dateizugriff und Dateisperren
SORT.EXE	sortiert Textdateien
SUBST.EXE	erzeugt ein virtuelles Laufwerk für einen angegebenen Pfadnamen.
SYS.COM	überträgt den Betriebssystem–Kern auf Disk
TREE.EXE	zeigt Dateien und Unterverzeichnisse einer Disk an
XCOPY.EXE	kopiert Dateien und Unterverzeichnisse

Diese Liste von Dateien (ggf. in anderer Reihenfolge) bekommen Sie angezeigt, wenn Sie den DIR–Befehl (s.a. ***MS–DOS–Benutzerhandbuch***) auf die Systemdiskette anwenden.

Die beiden Dateien MSDOS.SYS und IO.SYS sind in der o.a. Liste mit einem Stern (*) gekennzeichnet, weil sie sogenannte verborgene Dateien sind, die bei einem DIR–Befehl nicht angezeigt werden.

Im nächsten Kapitel lernen Sie, wie Sie Ihren PC oder PC–AT mit Hilfe der CONFIG.SYS–Datei für Ihre Anwendungen konfigurieren können.

2. Systemkonfigurierung

2.1 Was ist eine Konfigurationsdatei?

In den meisten Fällen benötigt MS–DOS installationsspezifische Parameter, die vor dem Systemstart konfiguriert werden müssen. Diese Parameter werden mit Hilfe spezieller Konfigurationsbefehle, die in einer besonderen Datei, der CONFIG.SYS–Datei, zusammengefaßt sind, eingestellt.

Jedesmal, wenn Sie MS–DOS starten, durchsucht MS–DOS das Basisverzeichnis, aus dem heraus es gestartet wird, nach der Datei CONFIG.SYS und interpretiert die in ihr enthaltenen Befehle. Wird die Datei nicht gefunden, so ordnet MS–DOS den Systemparametern Standardwerte zu. Mit Hilfe der CONFIG.SYS–Datei ist es z.B. möglich, besonderen angeschlossenen Geräten spezielle Gerätetreiber zuzuordnen.

2.2 Die CONFIG.SYS–Datei erzeugen

Enthält Ihre MS–DOS–Systemdiskette noch keine CONFIG.SYS–Datei, so können Sie diese mit Hilfe z.B. des Zeileneditors EDLIN (s. Kapitel 8) erzeugen und auf der Systemdiskette im Basisverzeichnis speichern. Ebenso leicht können Sie natürlich eine bereits vorhandenen CONFIG.SYS–Datei für Ihre speziellen Anforderungen mit dem Editor ändern.

2.3 Die CONFIG.SYS–Befehle im einzelnen

Die folgenden Befehle können Sie in einer CONFIG.SYS–Datei benutzen:

BREAK	schaltet die Überwachung der Ctrl–Break–Taste ein.
BUFFERS	setzt die Anzahl von Disk–Puffern im Hauptspeicher.
COUNTRY	setzt die Schreibweise für das landesspezifische Datums– und Zeitformat sowie Währungssymbol und Dezimalzeichen.
DEVICE	Installiert einen angegebenen Gerätetreiber im geladenen Betriebssystem.

DRIVPARM — setzt Parameter für Block–Geräte (Festplatten, Diskettenlaufwerke, Bandlaufwerke).

FCBS — setzt die Anzahl von Dateisteuerblöcken, die gleichzeitig geöffnet sein können.

FILES — setzt die Anzahl von gleichzeitig geöffneten Dateien, auf die MS–DOS zugreifen kann.

LASTDRIVE — setzt die maximale Anzahl von Laufwerken, auf die MS–DOS zugreifen kann.

SHELL — setzt den Namen des Befehlsprozessors, der anstelle von COMMAND.COM benutzt werden soll.

Diese Befehle werden im folgenden detailliert beschrieben. Ein Beispiel für eine CONFIG.SYS–Datei finden Sie im Abschnitt 2.4.

Vereinbarung der Notation

Im folgenden werden alle Befehle für die CONFIG.SYS–Datei in einem einheitlichen Format beschrieben. Dies gilt auch für die Befehlsbeschreibungen in den Kapitel 6, 8 und 11. Die Beschreibungen gliedern sich in:

Format: Hier ist die exakte Schreibweise für einen Befehl angegeben, dabei gelten die folgenden Syntax–Regeln:

1. Alle Wörter in Großbuchstaben sind Befehls–Schlüsselwörter, deren Schreibweise wie angegeben verbindlich ist. Bei der Eingabe über Tastatur können dabei Klein– oder Großbuchstaben verwendet werden, da MS–DOS diese Schlüsselwörter generell in Großbuchstaben umwandelt.

2. Alle Angaben in kursiver Groß– und Kleinschrift stellen Parameter dar, die vom Anwender eingesetzt werden müssen.

3. Angaben zwischen eckigen Klammern ([ ]) sind wahlfrei.

4. Ein senkrechter Strich (|) trennt Alternativparameter. Es kann entweder der Parameter vor oder der hinter dem senkrechten Strich angegeben werden.

5. Eine Folge von Punkten (. . .) hinter einer Angabe bedeutet, daß diese Angabe, falls erforderlich, mehrmals spezifiziert werden kann.

6. Alle Sonderzeichen außer den eckigen Klammern und dem senkrechten Strich, also , ; – = ( ) ' , müssen da, wo sie stehen, auch angegeben werden.

Bedeutung: Hier wird die Wirkung des jeweiligen Befehls beschrieben und die Parameter und ihre Definitionsbereiche werden erläutert.

Beispiele: Ein oderer mehrere Beispiele zeigen die Benutzung des Befehls.

Anmerkungen: Besondere Eigenschaften der einzelnen Befehle werden hier angegeben.

BREAK

Format: **BREAK=[ON|OFF]**

Bedeutung: Die Überwachung der Ctrl–Break–Tastenkombination zum Abbruch eines laufenden Programmes wird ein– (**ON**) oder ausgeschaltet (**OFF**).

Beispiel: `break=off`

Die Überwachung der Ctrl–Break–Taste durch MS–DOS während der Ausführung eines Programmes wird ausgeschaltet.

Anmerkungen:

- Bei den meisten Programmen kann der Programmablauf durch gleichzeitiges Drücken der beiden Tasten Ctrl und Break abgebrochen werden, wenn MS–DOS von der Tastatur liest oder auf den Bildschirm oder den Drucker schreibt. Mit BREAK=ON können zusätzlich Disk–Schreib–Lese–Operationen abgebrochen werden.

BUFFERS

Format: **BUFFERS = *n***

Bedeutung: Setzt die Anzahl von Disk–Puffern im Hauptspeicher beim Systemstart.

n Eine ganze Zahl zwischen 1 und 255. Voreingestellt ist ein Wert von 2.

Beispiel:

```
buffers=10
```

Beim Systemstart werden 10 Ein–/Ausgabepuffer für Disk–Schreib–/Leseoperationen im Hauptspeicher eingerichtet.

Anmerkungen:

- Ein Disk–Schreib–/Lesepuffer ist ein Speicherbereich, in dem MS–DOS die Daten, die von der Disk gelesen wurden oder auf die Disk geschrieben werden sollen, zwischenspeichert, wenn diese Daten nicht ein Vielfaches der Disk–Sektorgröße von 512 Bytes ausmachen.

- Bei Anwendungen, die intensive Disk–Schreib–Lese–Operationen ausführen, wie z.B. Text– oder Datenbankprogramme, empfiehlt sich zur Erhöhung der Verarbeitungsgeschwindigkeit ein Wert zwischen 10 und 20, bei der Verwendung vieler Disk-Unterverzeichnisse ein Wert zwischen 20 und 30.

COUNTRY

Format: **COUNTRY=*n***

Bedeutung: Setzt die Schreibweise für das landesspezifische Datums– und Zeitformat sowie Währungssymbol und Dezimalzeichen.

n Eine dreistellige Zahl, die den Ländercode (Telefonvorwahl) darstellt. Folgende Werte für ***n*** sind erlaubt:

n	**Land**
001	USA
031	Niederlande
032	Belgien
033	Frankreich
034	Spanien
039	Italien
041	Schweiz
044	Großbritannien
045	Dänemark
046	Schweden
047	Norwegen
049	Deutschland
061	Australien
358	Finnland
972	Israel

Beispiel:

```
country=049
```

Beim nächsten Start von MS–DOS wird das Datumsformat TT.MM.JJJJ, das Zeitformat HH.MM.SS, das Währungssymbol DM und das Dezimalzeichen , gesetzt.

DEVICE

Format: **DEVICE** = [***Laufw***:][***Pfad***]***Datei***[.***Erw***]

Bedeutung: Installiert einen Gerätetreiber, der sich in einer angegeben Datei im angegebenen Laufwerk befindet, im geladenen Betriebssystem.

Laufw: Eine nach MS–DOS–Konventionen gültige Laufwerksbezeichnung.

Pfad Ein Name für den Pfad (s.a. Kapitel 4.2.1), der zur Gerätetreiberdatei führt.

Datei Ein nach MS–DOS–Konventionen gültiger Dateiname (s.a. Kapitel 4.1.5).

Erw Eine nach MS–DOS–Konventionen gültige Dateinamenerweiterung (s.a. Kapitel 4.1.5).

Beispiel:

```
device=ansi.sys
```

Tastatureingabe und Bildschirmausgabe werden bei diesem Einheitentreiber mit der erweiterten Steuerung mit Hilfe der ANSI–Escape–Sequenzen unterstützt (s.a. Anhang B).

Anmerkungen:

- Die standardmäßig bei MS–DOS installierbaren Einheitentreiber sind ANSI.SYS, DRIVER.SYS und RAMDRIVE.SYS (s. Kapitel 2.5).

- Neben diesen Gerätetreibern kann der Anwender mit dem DEVICE–Befehl auch jederzeit eigene Gerätetreiber installieren.

DRIVPARM

Format: **DRIVPARM = /D:*ddd* [/C] [/F:*f*] [/H:*hh*] [/S:*ss*] [/T:*ttt*]**

Bedeutung: Setzt Parameter für jeweils ein Block–Geräte–Laufwerk (Festplatte, Diskette, Band), die die von den Original–MS–DOS–Gerätetreibern gesetzten überschreiben.

/D:*ddd* Nach dem Schalter /D kann eine logische Laufwerksnummer zwischen 0 und 255 angegeben werden, die den entsprechenden Laufwerksbuchstaben ersetzt (also A = 0, B = 1, C = 2 usw.).

/C Der /C–Schalter gibt an, daß die Laufwerksverriegelung unterstützt wird.

/F:*f*] Nach dem /F–Schalter kann ein Formfaktor für die Speicherkapazität des Mediums zwischen 0 und 7 mit folgender Bedeutung angegeben werden:

f	**Bedeutung**
0	320/360 KBytes
1	1,2 MBytes
2	720 KBytes
3	8 Zoll einfache Dichte
4	8 Zoll doppelte Dichte
5	Festplatte
6	Bandgerät
7	sonstiges Laufwerk

Voreingestellt ist für ***f*** der Wert 2 für ein Diskettenlaufwerk mit 720 KBytes.

/H:*hh* Nach dem /H–Schalter kann ein Wert zwischen 1 und 99 für die Anzahl der Schreib–/Lese–Köpfe angegeben werden. Wird dieser Schalter nicht gesetzt, so nimmt DRIVPARM den Wert 1 an.

/N Der /N–Schalter spezifiziert ein Blockgerät mit fest installiertem Datenträger, der nicht wechselbar ist.

/S:*ss* Nach dem /S–Schalter kann ein Wert zwischen 1 und 99 für die Anzahl der Sektoren pro Spur angegeben werden.

/T:*ttt* Nach dem /T–Schalter kann ein Wert zwischen 1 und 999 für die Anzahl der Spuren pro Seite für das Block–Speichermedium angegeben werden.

Beispiel:

Nehmen Sie an, Sie haben einen Computer mit einem eingebauten Band–Laufwerk als Einheit D:, das zur Systemstartzeit folgendermaßen konfiguriert sei: 20 Spuren mit je 40 Sektoren. Wenn Sie dieses Laufwerk nun so konfigurieren möchten, daß es 10 Spuren mit je 99 Sektoren schreibt, so müssen Sie folgenden Befehl in Ihre CONFIG.SYS–Datei einfügen:

```
drivparm=/d:3/f:6/h:1/s:99/t:10
```

Bei diesem Beispiel wird vorausgesetzt, daß der Gerätetreiber für das Bandlaufwerk diese Spur/Sektor–Konfiguration auch unterstützt.

FCBS

Format: **FCBS = *m,n***

Bedeutung: Setzt die Anzahl von Dateisteuerblöcken, die gleichzeitig geöffnet sein können.

m Ein ganzzahliger Wert zwischen 1 und 255, der die Anzahl gleichzeitig von Dateisteuerblöcken eröffenbarer Dateien angibt. Voreingestellt ist der Wert 4.

n Ein ganzzahliger Wert zwischen 1 und 255, der die Anzahl von durch Dateisteuerblöcke eröffneten Dateien angibt, die MS–DOS nicht automatisch schließen kann, wenn ein Anwenderprogramm mehr als ***m*** Dateien zu öffnen versucht. D.h., die ersten ***n*** durch Dateisteuerblöcke eröffneten Dateien werden gegen automatisches Schließen geschützt. Voreingestellt ist hier der Wert 0.

Beispiel:

```
fcbs=4,2
```

MS–DOS erlaubt das Öffnen von maximal 4 Dateien durch Dateisteuerblöcke, von denen die ersten 2 gegen automatisches Schließen geschützt sind, wenn mehr als 4 Dateien geöffnet werden sollten.

Anmerkungen:

- ***n*** darf nicht größer als ***m*** sein.
- Um Fehlerbedingungen zu vermeiden, die durch unbeabsichtigtes Schließen von Dateisteuerblöcken entstehen können, sollte der Wert ***m*** ausreichend groß gewählt werden.

FILES

Format: **FILES = *n***

Bedeutung: Setzt die Anzahl der maximal für MS–DOS–Zugriffe eröffenbaren Dateien.

n Ein ganzzahliger Wert zwischen 8 und 255. Voreingestellt ist ein Werte von 8. Werte über 20 haben keine Funktion.

Beispiel:

```
files=20
```

Es können maximal 20 Dateien eröffnet werden, auf die über MS–DOS–Systemaufrufe zugegriffen werden kann.

Anmerkungen:

- Wenn MS–DOS über eine Dateinummer mitgeteilt wird, daß eine Datei für Zugriffe eröffnet werden soll, so legt es in einem geschützen Bereich dafür einen separaten Dateisteuerblock an. Die Größe dieses Bereiches wird von dem Wert ***n*** im Files–Befehl festgelegt.

- Die MS–DOS–Systemaufrufe 2FH bis 60H sind mit dem Betriebssystem XENIX kompatibel.

LASTDRIVE

Format: **LASTDRIVE = *x***

Bedeutung: Setzt die maximale Anzahl Laufwerke, auf die zugegriffen werden kann.

x Ein beliebiger Buchstabe des Alphabets. Voreingestellt ist E, unabhängig von der Anzahl der tatsächlich installierten Laufwerke.

Beispiel:

```
lastdrive=f
```

Es kann auf insgesamt 6 Laufwerke zugegriffen werden.

Anmerkungen:

- Der Mindestwert für ***x*** muß der Anzahl der installierten Laufwerke entsprechen. Ist ***x*** kleiner, so wird der LASTDRIVE–Befehl in der CONFIG.SYS–Datei ignoriert.

- Dieser Befehl ist nur in einer Netzwerk–Umgebung sinnvoll. Das Netzwerk muß neu initialisiert werden, um alle mit LASTDRIVE definierten Laufwerke für gültig zu erklären.

- Da MS–DOS für jedes angegebene Laufwerk eine eigene Datenstruktur anlegt, die Hauptspeicherplatz verbraucht, sollten nicht mehr Laufwerke als nötig angegeben werden.

SHELL

Format: **SHELL** = [***Laufw***:][***Pfad***]***Datei***[.***Erw***]
[**/E:*n***] [***Laufw***:][***Pfad***] [**/P**]

Bedeutung: Setzt den Namen des Befehlsprozessors, der anstelle von COMMAND.COM von der angegebenen Datei geladen werden soll.

Laufw: Eine nach MS–DOS–Konventionen gültige Laufwerksbezeichnung.

Pfad Ein Name für den Pfad (s.a. Kapitel 4.2.1), der zur Befehlsprozessordatei führt bzw. der im zweiten Fall angibt, von wo der Befehlsprozessor ggf. erneut geladen werden soll.

Datei Ein nach MS–DOS–Konventionen gültiger Dateiname (s.a. Kapitel 4.1.5).

Erw Eine nach MS–DOS–Konventionen gültige Dateinamenerweiterung (s.a. Kapitel 4.1.5).

/E:*n* Der /E–Schalter setzt den Speicher für den Befehlsprozessor auf ***n*** Bytes.

/P Der /P–Schalter bedeutet, daß der angegebene Befehlsprozessor das erste auf dem System laufende Programm ist, so daß der EXIT–Befehl (s. Kapitel 6) zur Rückkehr auf die Befehlsprozessor–Grundebene ausgeführt werden kann.

Beispiel:

```
shell=\bin\comint.com /e:3000 a:\bin /p
```

Der anwenderspezifische Befehlsprozessor COMINT.COM wird aus dem Unterverzeichnis BIN im Standardlaufwerk geladen. Es wird dafür ein Speicher von 3000 Bytes reserviert, COMINT.COM soll bei Bedarf aus dem Unterverzeichnis BIN im Laufwerk A: nachgeladen werden und ist das erste auf dem System laufende Programm.

Anmerkungen:

- Diesen Befehl sollten nur erfahrene Programmierer verwenden, die ihren eigenen Befehlsprozessor schreiben wollen. Dabei müssen Unterbrechungs– und Stapelverarbeitungsbefehle aus COMMAND-.COM übernommen werden.

2.4 Ein Beispiel

Eine typische Konfigurationsdatei CONFIG.SYS könnte so aussehen:

```
buffers=10
files=10
device=\bin\network.sys
break=on
shell=a:\bin\comint.com /e:3000 a:\bin /p
lastdrive=e
country=049
```

Es werden Diskpuffer im Hauptspeicher angelegt und es können maximal 10 Dateien gleichzeitig geöffnet sein. Der Netzwerk–Gerätetreiber NETWORK-.SYS soll aus dem Unterverzeichnis BIN im Standardlaufwerk geladen werden. Die Überwachung der Ctrl–Break–Taste soll auch für Disk–Operationen aktiviert werden.

Ein anwenderspezifischer Befehlsprozessor COMINT.COM soll aus dem Unterverzeichnis BIN im Standardlaufwerk geladen werden, für den ein Speicher von 3000 Bytes reserviert werden soll, der aus A:BIN ggf. nachgeladen werden kann und als erstes Programm im System läuft.

Es soll auf maximal 5 Laufwerke zugegriffen werden und die deutsche Datums–, Uhrzeit–, Währungssymbol– und Dezimalzeichen–Schreibweise soll aktiviert werden.

2.5 Installierbare Gerätetreiber

Mit der MS–DOS–Systemdiskette werden drei installierbare Gerätetreiber, ANSI.SYS, DRIVER.SYS und RAMDRIVE.SYS, mitgeliefert. Im folgenden werden Einzelheiten zu diesen drei Treibern beschrieben.

2.5.1 ANSI.SYS

Eine ANSI–ESCAPE–Sequenz ist eine Folge von Zeichen, beginnend mit dem ESC–Zeichen (1BH), mit der ganz allgemein Steuerfunktionen für MS–DOS definiert werden können. Insbesondere werden damit grafische Funktionen sowie Steuerfunktionen für den Cursor (Positionsanzeiger auf dem Bildschirm) definiert. Zur Installation dieses Treibers müssen Sie folgenden Befehl in Ihre CONFIG.SYS–Datei einfügen:

DEVICE = ANSI.SYS

Hinweis: Die ESCAPE–Sequenzen können von einem Anwenderprogramm über die MS–DOS–Systemaufrufe 1, 2, 6 und 9 ausgegeben werden und setzen wie o.a. die Installation des Bildschirm– und Tastaturtreibers ANSI-.SYS voraus.

Im folgenden werden die bei installiertem Bildschirm– und Tastaturtreiber ANSI.SYS verfügbaren Funktionen, die über ESCAPE–Sequenzen aufgerufen werden, beschrieben. Bei den einzelnen Beschreibungen werden die folgenden Abkürzungen verwendet:

ESC repräsentiert das Zeichen mit dem Code 1BH und **nicht** die drei Buchstaben ESC.

Pn repräsentiert einen numerischen Parameter, für den eine Dezimalzahl in Form von ASCII–Ziffern angegeben werden muß.

Ps repräsentiert einen selektiven Parameter, ebenfalls eine Zahl, mit dem eine Unterfunktion gewählt werden kann. Es können mehrere Unterfunktionen angegeben werden, die dann durch Semikolon (;) voneinander getrennt werden müssen.

Pz repräsentiert den Zeilenparameter; eine Dezimalzahl, die in Form von ASCII–Ziffern angegeben werden muß.

Ps repräsentiert den Spaltenparameter; eine Dezimalzahl, die in Form von ASCII–Ziffern angegeben werden muß.

Wenn kein Parameter oder Null angegeben wird, verwendet MS–DOS einen voreingestellten Wert.

2.5.1.1 Cursor–Steuerfunktionen

CUP – Cursor–Position

Format: **ESC[*Pz*;*Ps*H**

Bedeutung: Plaziert den Positionszeiger für die Texteingabe auf dem Bildschirm in die Zeile ***Pz*** bei der Spalte ***Ps***. Der voreingestellte Wert für beide Parameter ist 1. Wird kein Parameter angegeben, so wird der Cursor in die Home-Position (d.i. die äußerst linke Spalte in der ersten Bildschirmzeile) gesetzt.

HVP – Horizontal– und Vertikal–Position

Format: **ESC[*Pz*;*Ps*F**

Bedeutung: Plaziert den Positionszeiger für die Texteingabe auf dem Bildschirm in die Zeile ***Pz*** bei der Spalte ***Ps***. Der voreingestellte Wert für beide Parameter ist 1. Wird kein Parameter angegeben, so wird der Cursor in die Home-Position (d.i. die äußerst linke Spalte in der ersten Bildschirmzeile) gesetzt.

CUU – Cursor nach oben

Format: **ESC[*Pn*A**

Bedeutung: Plaziert den Cursor in derselben Spalte ***Pn*** Zeilen oberhalb der aktuellen Zeile. Voreingestellt ist für ***Pn*** der Wert 1. Befindet sich der Cursor bereits in der obersten Bildschirmzeile, so wird dieser Funktionsaufruf ignoriert.

CUD – Cursor nach unten

Format: **ESC[*Pn*B**

Bedeutung: Plaziert den Cursor in derselben Spalte ***Pn*** Zeilen unterhalb der aktuellen Zeile. Voreingestellt ist für ***Pn*** der Wert 1. Befindet sich der Cursor bereits in der untersten Bildschirmzeile, so wird dieser Funktionsaufruf ignoriert.

CUF – Cursor nach rechts

Format: **ESC[*Pn*C**

Bedeutung: Plaziert den Cursor in derselben Zeile ***Pn*** Spalten nach rechts. Voreingestellt ist für ***Pn*** der Wert 1. Befindet sich der Cursor bereits in der äußerst rechten Spalte, so wird dieser Funktionsaufruf ignoriert.

CUF – Cursor nach links

Format: **ESC[*Pn*D**

Bedeutung: Plaziert den Cursor in derselben Zeile ***Pn*** Spalten nach links. Voreingestellt ist für ***Pn*** der Wert 1. Befindet sich der Cursor bereits in der äußerst linken Spalte, so wird dieser Funktionsaufruf ignoriert.

DSR – Geräte–Status–Report

Format: **ESC[6n**

Bedeutung: Der Konsol–Treiber gibt eine CPR–Sequenz (s.u.) aus, wenn er die DSR–ESCAPE–Sequenz empfängt.

CPR – Positionsanzeiger–Stellung melden

Format: **ESC[*Pn*;*Pn*R**

Bedeutung: Diese ESCAPE–Sequenz liefert einen Bericht über die aktuelle Cursor–Position über das Standard–Eingabegerät. Der erste Parameter bezeichnet die aktuelle Zeile und der zweite die aktuelle Spalte.

SCP – Positionsanzeiger–Stellung merken

Format: **ESC[s**

Bedeutung: Der Konsol–Treiber merkt sich die aktuelle Cursor–Position.

RCP – Positionsanzeiger–Stellung wiederherstellen

Format: **ESC[u**

Bedeutung: Die Cursor–Position wird auf den Wert wiederhergestellt, den sie hatte, als der Konsol–Treiber die SCP–Sequenz (s.o.) empfangen hat.

2.5.1.2 Lösch–Funktionen

ED – Bildschirm löschen

Format: **ESC[2J**

Bedeutung: Der gesamte Bildschirm wird gelöscht und der Cursor wird in die obere linke Bildschirmecke (Home–Position) gesetzt.

EL – Zeilenende löschen

Format: **ESC[K**

Bedeutung: Alle Zeichen von einschließlich der aktuellen Cursorposition bis zum Ende der Bildschirmzeile werden gelöscht.

2.5.1.3 Grafische Funktionen

Die folgenden ESCAPE–Sequenzen betreffen die Bildschirmgrafik, setzen jedoch einen Farb–/Grafikadapter voraus.

SGR – Grafikattribute setzen

Format: **ESC[*Ps*;. . .;*Ps*m**

Bedeutung: Diese Sequenz ruft die grafischen Funktionen auf, die durch die Parameter ***Ps*** angegeben werden. Ein gesetztes Attribut bleibt bis zur nächsten SGR–Sequenz wirksam. Folgende Funktionen sind verfügbar:

Ps	**Funktion**
0	Alle Attribute aus (weiße Schrift auf schwarzem Hintergrund)
1	halbfett ein
2	matt ein
3	kursiv ein
5	Blinken ein
6	schnelles Blinken ein
7	inverse Darstellung ein
8	unsichtbare Darstellung ein
30	schwarzer Vordergrund
31	roter Vordergrund
32	grüner Vordergrund
33	gelber Vordergrund
34	blauer Vordergrund
35	violetter Vordergrund
36	kobaltblauer Vordergrund
37	weißer Vordergrund
40	schwarzer Hintergrund
41	roter Hintergrund
42	grüner Hintergrund
43	gelber Hintergrund
44	blauer Hintergrund
45	violetter Hintergrund
46	kobaltblauer Hintergrund
47	weißer Hintergrund
48	Tiefstellen ein (Indices)
49	Hochstellen ein (Exponenten)

Die Parameter 30 bis 47 entsprechen dem ISO6429-Standard.

SM – Modus setzen

Formate:
ESC[=*Ps*h
ESC[=h
ESC[=0h
ESC[?7h

Bedeutung: Ändert die Breite der Bildschirmanzeige oder die Art der Bildschirmanzeige. Folgende Parameter ***Ps*** sind verfügbar:

Ps	**Funktion**
0	Text 40 Spalten x 25 Zeilen schwarz und weiß
1	Text 40 Spalten x 25 Zeilen Farbe
2	Text 80 Spalten x 25 Zeilen schwarz und weiß
3	Text 80 Spalten x 25 Zeilen Farbe
4	Grafik 320 x 200 Punkte Farbe
5	Grafik 320 x 200 Punkte schwarz und weiß
6	Grafik 640 x 200 Punkte schwarz und weiß
7	automatischer Zeilenwechsel bei Texteingabe über das Zeilenende hinaus.

RM – Modus rücksetzen

Formate:
ESC[=*Ps*l
ESC[=l
ESC[=0l
ESC[?7l

Bedeutung: Die Parameter sind hier dieselben wie für die Funktion SM (s.o.) mit Ausnahme des Parameters 7, mit dem hier der automatische Zeilenüberlauf wieder abgeschaltet wird. Die bei der Texteingabe über das Zeilenende hinausgeschriebenen Zeichen gehen also verloren.

2.5.1.4 Umdefinition der Tastaturbelegung

Zusätzlich erlaubt ANSI.SYS die Ausgabe von besonderen ESCAPE–Sequenzen, mit denen die Belegung einzelner Tasten der Tastatur umdefiniert werden kann.

Formate:

ESC[*Pn*;*Pn*;. . .;*Pn*p

ESC["*Zeichenkette*";p

ESC[*Pn*;"*Zeichenkette*";*Pn*;"*Zeichenkette*";*Pn*p

Jede weitere Kombination von Dezimalzahlen und Zeichenkette

Bedeutung:

Die erste angegebene Zahl ***Pn*** definiert den ASCII–Code, dessen zugeordnete Taste umdefiniert werden soll. Die restlichen Zahlen ***Pn*** definieren die Codes der Zeichen, die angezeigt werden, wenn die entsprechende Taste gedrückt wird. Ist der erste Code Null, so bezeichnen der erste **und** zweite Code eine der Steuer– oder Funkionstasten der Tastatur, die mit der nachfolgenden Codefolge belegt werden soll. Im *Technischen Handbuch* sind die den einzelnen Tasten der Tastatur zugeordneten Codes angegeben.

Beispiele:

Durch die folgenden ESC–Sequenzen werden die A–Taste für Groß– und Kleinschreibung mit Q und q belegt und die Q–Taste für Groß– und Kleinschreibung mit A und a:

ESC[65;81p	A wird Q
ESC[97;113p	a wird q
ESC[81;65p	Q wird A
ESC[113;97p	q wird a

Die Funktionstaste F10 soll mit dem Text DIR und anschließendem Wagenrücklaufcode (Eingabe–Taste) belegt werden:

ESC[0;68;"dir";13p

Die beiden Codes 0 und 68 bezeichnen die Funktionstaste F10, der Code 13 ist der Eingabetaste zugeordnet-

2.5.2 DRIVER.SYS

Dieser Treiber unterstützt beliebige extern angeschlossene Laufwerke. Zur Installation müssen Sie folgenden Befehl in Ihre CONFIG.SYS–Datei einfügen:

DEVICE = DRIVER:SYS /D:*ddd* [/C] [/F:*f*] [/H:*hh*]
[/N] [/S:*ss*] [/T*ttt*]

Dabei bedeuten:

/D:*ddd* Nach dem Schalter /D kann eine logische Laufwerksnummer zwischen 0 und 255 angegeben werden. Physikalische Laufwerke werden anders als logische Laufwerke numeriert. Diskettenlaufwerke werden beginnend mit 00H und Festplattenlaufwerke beginnend mit 80H numeriert.

Wenn Sie z.B. einen Computer mit zwei Diskettenlaufwerken haben, so werden diese mit 00H und 01H numeriert. Ein weiteres erhält die Nummer 02H. Verfügen Sie dagegen über einen Computer mit einem Disketten– und einem Festplattenlaufwerk, so erhält das Diskettenlaufwerk die Nummer 00H, die Festplatte dagegen 80H. Schließen Sie jetzt ein weiteres externes Diskettenlaufwerk an Ihr System an, so erhält dieses die Nummer 02H, da MS–DOS die Kennungen für die physikalischen Laufwerke 00H und 01H bereits kennt.

/C Der /C–Schalter gibt an, daß Laufwerksverriegelung unterstützt wird.

/F:*f* Nach dem /F–Schalter kann ein Formfaktor für die Speicherkapazität des Mediums zwischen 0 und 7 mit folgender Bedeutung angegeben werden:

f	**Bedeutung**
0	320/360 KBytes
1	1,2 MBytes
2	720 KBytes
3	8 Zoll einfache Dichte
4	8 Zoll doppelte Dichte
5	Festplatte
6	Kassetten–Bandgerät
7	sonstiges Laufwerk

Voreingestellt ist für ***f*** der Wert 2 für ein Diskettenlaufwerk mit 720 KBytes.

/H:*hh* Nach dem /H–Schalter kann ein Wert zwischen 1 und 99 für die Anzahl der Schreib–/Lese–Köpfe angegeben werden.

/N Der /N–Schalter spezifiziert ein Blockgerät mit fest installiertem Datenträger, der nicht wechselbar ist.

/S:*ss* Nach dem /S–Schalter kann ein Wert zwischen 1 und 99 für die Anzahl der Sektoren pro Spur angegeben werden.

/T:*ttt* Nach dem /T–Schalter kann ein Wert zwischen 1 und 999 für die Anzahl der Spuren pro Seite für das Block–Speichermedium angegeben werden.

Beispiel: Nehmen Sie an, Sie wollen an Ihr System ein Diskettenlaufwerk mit 720 KBytes anschließen. Dann müssen Sie folgenden Befehl in Ihre CONFIG.SYS–Datei einfügen:

```
device=driver.sys /d:02
```

2.5.3 RAMDRIVE.SYS

RAMDRIVE.SYS ist ein Gerätetreiber, der es Ihnen erlaubt, einen Teil des Hauptspeichers Ihres Computers als ein Disklaufwerk zu organisieren und als ein solches zu benutzen. Ein solcher Speicherbereich wird als RAM–Disk oder als virtuelle Disk bezeichnet.

Eine RAM–Disk erlaubt einen wesentlich schnelleren Datenzugriff als eine normale Disk, da jegliche Mechanik entfällt und die Information bereits im Hauptspeicher zur Verfügung steht. Wenn Ihr Computer mit ausreichend Speicher (mehr als 1 MByte) oder einer Speichererweiterung nach dem LOTUS–INTEL–MICROSOFT–Standard ausgerüstet ist, können Sie Teile dieses Speichers für ein oder mehrere RAM–Disk–Laufwerke verwenden. Andernfalls würde der Gerätetreiber RAMDRIVE.SYS die RAM–Disk im unteren Speicherbereich anlegen.

Beachten Sie: Auf jeden Fall vergrößert der Gerätetreiber RAMDRIVE.SYS den speicherresidenten Teil von MS–DOS.

Zur Installation einer RAM–Disk im Hauptspeicher Ihres Computers verwenden Sie den folgenden Befehl:

DEVICE = RAMDRIVE.SYS [***bbb***] [***sss***] [***ddd***] [**/E**] [**/A**]

Dabei bedeuten:

bbb Die Disk–Kapazität in Kilobytes. Voreingestellt sind 64 KBytes. Der kleinste erlaubte Wert ist 16 KBytes.

sss Die Sektorgröße in Bytes. Voreingestellt sind 128 Bytes. Erlaubte Werte sind 128, 256, 512, 1024.

ddd Die maximale Anzahl von Einträgen im Basis–Verzeichnis der RAM-Disk. Voreingestellt ist 64. Der kleinste erlaubte Wert ist 2, der größte 1024.

RAMDRIVE.SYS rundet den ***ddd***–Wert nach der nächst höheren Sektor-Grenze auf. Wenn Sie für ***ddd*** z.B. einen Wert von 10 bei einer gewählten Sektorgröße von 128 Bytes angeben, so wird 10 auf 12 aufgerundet, da 12 Einträge zu je 32 Bytes ein Vielfaches der gewählten Sektorgröße sind.

/E Dieser Schalter erlaubt die Nutzung von einem ggf. installierten weiteren Speicher oberhalb der 1–MBytes–Grenze. Wenn Sie den /E–Schalter benutzen, können Sie nicht den /A–Schalter (s.u.) benutzen.

/A Dieser Schalter erlaubt die Nutzung des zusätzlichen Speichers auf einer ggf. installierten Erweiterungskarte nach dem LOTUS–INTEL–MICROSOFT–Standard. Wenn Şie den /A–Schalter benutzen, können Sie nicht den /E–Schalter (s.o.) benutzen.

Beispiel: Nehmen Sie an, Sie wollen in Ihrem mit 640 KBytes Hauptspeicher ausgerüsteten Computer eine RAM–Disk mit 128 KBytes mit der Standardsektorgröße von 128 Bytes für maximal 64 Verzeichniseinträge einrichten. Dann müssen Sie in Ihre CONFIG.SYS–Datei folgenden Befehl einfügen:

```
device=ramdrive.sys 128
```

3. Festplatten–Konfigurierung mit FDISK

3.1 Einleitung

Wenn Ihr PC über eine Festplatte verfügt, so muß diese vor der ersten Benutzung mit MS–DOS konfiguriert werden, andernfalls erhalten Sie eine Fehlermeldung.

Eine Festplatte kann unter MS–DOS in ein bis vier Bereiche, sogenannten Partitionen, unterteilt werden. Solche Partitionen sind als von einander unabhängige Plattenbereiche zu verstehen, die jeder für sich ein anderes Betriebssystem enthalten können.

Um die Festplatte mit MS–DOS benutzen zu können, müssen Sie dafür eine Partition erzeugen, die DOS–Partition. Für diesen Zweck enthält Ihre Systemdiskette einen speziellen MS–DOS–Befehl, FDISK, der Ihnen bildschirmmenü–geführt diese Arbeit abnimmt. Den Befehl FDISK benötigen Sie für folgende Aufgaben:

- Erzeugen einer DOS–Partition
- Wechseln der aktuellen Partition
- Löschen einer DOS–Partition
- Informationen über vorhandene Partitionen anzeigen
- Nächstes Festplattenlaufwerk für die Partitionierung auswählen.

Wenn Sie, wie das bei den meisten Anwendungen der Fall sein wird, die gesamte Festplatte für MS–DOS benutzen wollen, benötigen Sie den FDISK–Befehl nur ein einziges Mal, um die DOS–Partition zu erzeugen.

Beachten Sie: Viele Computer–Händler konfigurieren Festplatten–Computer bereits für MS–DOS, so daß der Kunde FDISK nicht mehr benutzen muß. Meistens wird die Festplatte auch schon formatiert und MS–DOS wird auf die Festplatte übertragen, so daß das System nach dem Einschalten automatisch von der Festplatte gestartet wird. Um dieses festzustellen, brauchen Sie nur zu versuchen, Ihren Rechner ohne eingelegte Systemdiskette zu starten oder MS–DOS von der mitgelieferten Systemdiskette zu starten, den

FDISK–Befehl aufzurufen und zu versuchen, eine DOS–Partition zu erzeugen. Ist diese bereits vorhanden, so erhalten Sie eine entsprechende Meldung.

3.2 FDISK starten

Um FDISK zu starten, brauchen Sie nur

```
fdisk
```

einzugeben und die Eingabetaste zu drücken. Sie erhalten dann das Hauptmenü von FDISK auf dem Bildschirm angezeigt (im folgenden werden die Bildschirm–Meldungen von FDISK nur im Original, also amerikanisch, angegeben):

```
Fixed Disk Setup Program Version 0.02
(C)Copyright Microsoft, 1985.

FDISK Options

Choose one of the following:

    1. Create DOS Partition
    2. Change Active Partition
    3. Delete DOS Partition
    4. Display Partition Data
    5. Select Next Fixed Disk Drive

Enter choice:[1]

Press ESC to return to DOS
```

Der Menü–Punkt 5 wird nicht angezeigt, wenn Ihr Computer nur über ein Festplattenlaufwerk verfügt.

Die folgenden Abschnitte dieses Kapitel enthalten jeweils die Beschreibung eines der o.a. Menüpunkte zusammen mit allen Bildschirmanzeigen.

Um vom FDISK–Hauptmenü zurück zur MS–DOS–Befehlsebene zu gelangen, brauchen Sie nur die ESC–Taste zu drücken. Mit dieser Taste gelangen Sie auch aus jedem FDISK–Untermenü zurück ins Hauptmenü. Bei jedem Menü wird eine voreingestellte Auswahl angezeigt. Für diese Auswahl

braucht nur die Eingabetaste gedrückt zu werden, andernfalls wird der gewünschte Wert eingegeben und dann die Eingabetaste gedrückt.

3.2.1 Eine DOS–Partition einrichten

Wenn Sie den ersten Menüpunkt im FDISK–Hauptmenü durch Drücken der Eingabetaste gewählt haben und Ihre Festplatte enthält bereits eine DOS-Partition, so zeigt FDISK über diese Partition eine Statusinformation an. Z.B. sieht Ihr Bildschirm dann so aus:

```
Create DOS Partition

Partition  Status   Type   Start  End   Size
    1         A      DOS     0    613   614

Total disk space is 614 cylinders

Fixed disk already has a DOS partition

Press ESC to return to FDISK options
```

Enthält Ihre Festplatte dagegen keine DOS–Partition, so zeigt FDISK folgendes an:

```
Create DOS Partition

Current Fixed Disk Drive: 1

Do you wish to use the entire fixed disk
for DOS (Y/N).............?[Y]
```

Wenn Sie die gesamte Festplatte für MS–DOS benutzen wollen, was in der Regel der Fall ist, brauchen Sie nur die Eingabetaste zu drücken, um die Voreinstellung Y (Yes=Ja) zu akzeptieren. FDISK zeigt dann folgende Meldung:

```
System will now restart
Insert DOS diskette in drive A:
Press any key when ready...
```

Jetzt müssen Sie die MS–DOS–Systemdiskette in Laufwerk A einlegen und eine beliebige Taste drücken, um MS–DOS neu zu starten.

Nach erfolgter Partitionierung müssen Sie Ihre Festplatte formatieren, damit MS–DOS sie benutzen kann. Wenn Sie zukünftig MS–DOS von der Festplatte starten wollen, was wegen der wesentlich größeren Geschwindigkeit zu empfehlen ist, sollten Sie den FORMAT–Befehl mit dem /S–Schalter (s.a. bei FORMAT im Kapitel 6.3) benutzen.

Wenn Sie nur einen Teil der Festplatte für MS–DOS benutzen wollen, müssen Sie bei der o.a. Anforderung

```
Do you wish to use the entire fixed disk
for DOS (Y/N).....? [Y]
```

die N–Taste und dann die Eingabetaste drücken. FDISK zeigt dann folgende Meldung an:

```
Total fixed disk space is xxx cylinders
Maximum available space is xxx
cylinders at xxx
Enter partition size.....: [xxx]
```

Bei dieser Meldung wird die insgesamt für eine Festplattenpartition verfügbare Anzahl von Zylindern angezeigt und vom Benutzer die Eingabe der Größe der neuen Partition gefordert. Die voreingestellte Größe ist der maximal auf der Festplatte verfügbare Platz. Wenn Sie diese Größe wollen, drücken Sie einfach die Eingabetaste, andernfalls geben Sie die Größe in Zylindern ein und drücken dann die Eingabetaste. FDISK zeigt dann folgende Meldung:

```
Enter starting cylinder number. .:[xxx]
```

Beachten Sie hier: Wenn FDISK am Anfang einer Partition defekte Spuren findet, richtet es die Partitionsgrenzen neu aus, um die fehlerhaften Spuren zu übergehen.

Die voreingestellte Start–Zylinder–Nummer kennzeichnet den ersten Zylinder, der groß genug ist, um die Partition aufzunehmen. Wenn Sie die voreingestellte Startposition akzeptieren, drücken Sie nur die Eingabetaste, andernfalls geben Sie die Nummer des Zylinders an, ab dem Sie die Partition erzeugen wollen.

Wenn Sie später MS–DOS aus dieser Partition heraus starten wollen, vergessen Sie nicht die Angabe des /S–Schalters beim FORMAT–Befehl (s.a. bei FORMAT im Kapitel 6.3).

Beachten Sie: Sie können nur **eine** DOS–Partition auf Ihrer Festplatte einrichten. Wenn Sie auf dieser Platte weitere Partitionen für andere Betriebssysteme einrichten wollen, müssen Sie deren Dokumentation zu Rate ziehen.

3.2.2 Die aktuelle Partition ändern

Wenn Sie den zweiten Menüpunkt des FDISK–Hauptmenüs wählen, wird der Status jeder Partition auf Ihrer Festplatte angezeigt. Die aktive Partition, mit dem Statuswert A gekennzeichnet, ist diejenige Partition, auf deren Betriebssystem und Dateien Sie zugreifen, wenn Sie Ihren Computer einschalten oder rücksetzen. Zu einer Zeit kann immer nur eine Partition aktiv sein. Die anderen ggf. vorhandenen Partitionen sind inaktiv und mit dem Statuswert N gekennzeichnet.

Z.B. kann Ihr Bildschirm so aussehen, wenn Sie die aktive Partition ändern wollen:

```
Change aktive Partition

Current Fixed Disk Drive: 1

Partition  Status   Type   Start  End   Size
    1         A      DOS       0  306   307
    2         N  non-DOS     308  614   307

Total disk space ist 614 cylinders

Enter the number of the partition you
want to make aktive.............: [1]

Press ESC to return to FDISK options
```

Geben Sie jetzt die Nummer der zu aktivierenden Partition ein und drücken Sie dann die Eingabetaste. Voreingestellt ist die Nummer der aktiven Partition.

Wenn Sie die gesamte Festplatte als DOS–Partition nutzen, wird statt der Eingabeaufforderung die folgende Meldung angezeigt:

```
Partition 1 is already active
```

Um zum FDISK–Hauptmenü zurückzukehren, brauchen Sie nur die ESC–Taste zu drücken.

3.2.3 Eine DOS–Partition löschen

Wenn Sie den dritten Menüpunkt des Hauptmenüs wählen, zeigt FDISK den Status jeder Partition auf Ihrer Festplatte an und fordert Sie auf, die Löschung der DOS–Partition noch einmal zu bestätigen. Wenn Sie eine DOS–Partition löschen, löscht FDISK die Partitionsgrenzen und alle in dieser Partition existierenden Daten. **Die Daten können nach dem Löschen nicht wiederhergestellt werden.**

Beachten Sie: Mit FDISK können Sie keine Nicht–DOS–Partition löschen. Wenn Sie die DOS–Partition gelöscht haben, müssen Sie vielmehr eine MS–DOS–Systemdiskette in Laufwerk A einlegen. Wenn Sie dagegen ein anderes Betriebssystem in einer anderen Partition auf Ihrer Festplatte starten wollen, müssen Sie die Partition auf diese Nummer ändern, **ehe** Sie die DOS–Partition löschen.

Vor dem Löschvorgang zeigt FDISK z.B. diesen Bildschirm:

```
Delete DOS Partition

Current Fixed Disk Drive: 1

Partition  Status   Type   Start  End   Size
    1         A      DOS     0    613   614

Total disk space ist 614 cylinders

Warning! Data in the DOS partition
will be lost. Do you wish to
continue.............? [N]

Press ESC to return to FDISK options
```

Wenn Sie die DOS–Partition nicht löschen wollen, drücken Sie nur die Eingabetaste, um die voreingestellte Antwort N (NO = Nein) zu akzeptieren, andernfalls drücken Sie erst die Taste Y (Yes = Ja) und dann die Eingabetaste.

3.2.4 Partitionsinformationen anzeigen

Wenn Sie den vierten Menüpunkt des Hauptmenüs wählen, zeigt FDISK Informationen über jede Partition auf Ihrer Festplatte an.

Z.B. könnte der Bildschirm dann so aussehen:

```
Display Partition Information

Partition  Status   Type   Start  End   Size
    1         A      DOS     0    613   614

Total disk space is 614 cylinders

Press ESC to return to FDISK options
```

Die Information enthält Nummer, Status und Typ der Partition(en), Nummer von Start– und End–Zylinder sowie die Größe in Zylindern. Drücken Sie die ESC–Taste, um zum FDISK–Hauptmenü zurückzukehren.

3.2.5 Eine weitere Festplatte partitionieren

Dieser Menüpunkt erscheint nur dann im FDSIK–Hauptmenü, wenn mehr als eine Festplatte an Ihrem Computer angeschlossen ist. Wenn Sie diesen Punkt wählen, schaltet FDISK auf das nächste Laufwerk um.

Wenn z.B. das aktuelle Festplattenlaufwerk die Nummer 1 hat und Sie den Menüpunkt 5 auf dem FDISK–Hauptmenü wählen, aktiviert FDISK das Laufwerk Nummer 2. Dann können Sie wieder einen der Hauptmenüpunkte wählen, um das zweite Festplattenlaufwerk für MS–DOS vorzubereiten.

4. Dateien und Verzeichnisse

Ehe Sie dieses Kapitel durcharbeiten, sollten Sie wissen, wie Sie MS–DOS starten, wie Sie Disketten formatieren und duplizieren können, wie Sie Dateien kopieren und löschen können und wie Sie Programme ablaufen lassen können. Fehlen Ihnen diese Informationen noch, so sollten Sie zunächst das *MS–DOS–Bedienungshandbuch* zu Rate ziehen.

4.1 Dateien

4.1.1 Was ist eine Datei?

Unter einer Datei versteht man die Gesamtheit zusammengehöriger Informationen. Eine Datei auf einer Disk ist vergleichbar mit einem Aktenordner in einem Aktenschrank. Ein Aktenordner kann z.B. Geschäftsbriefe von Kunden oder wichtige Unterlagen zu einem Projekt enthalten.

Solche Informationen können aber auch genauso gut in einer Disk–Datei enthalten sein.

Alle Programme wie auch Texte oder sonstige Daten, die auf einer Disk gespeichert sind, sind in Dateien logisch organisiert und jede Datei hat einen einzigartigen Namen. Auf eine Datei wird über ihren Namen Bezug genommen. In diesem Abschnitt werden Sie deshalb zunächst lernen, wie man Dateien benennt.

Jedesmal, wenn Sie mit Ihrem Computer eingegebenen Text oder Daten abspeichern wollen, wird eine Datei erzeugt. Ebenso erzeugen Sie eine Datei, wenn Sie ein Programm geschrieben haben und dieses auf Disk speichern wollen.

4.1.2 Wie MS–DOS Dateien verwaltet

Die Namen aller Dateien einer Disk werden in Verzeichnissen auf der Disk gespeichert. Diese Verzeichnisse enthalten zusätzlich Informationen über die Dateigröße sowie den Zeitpunkt der Erzeugung bzw. letzten Fortschrei-

bung einer jeden Datei. Das Verzeichnis, das die gerade von Ihnen bearbeitete Datei enthält, wird als aktuelles Verzeichnis bezeichnet.

In einem weiteren Bereich der Disk, der sogenannten Dateizuordnungstabelle (file allocation table), verwaltet MS–DOS die Disk–Adressen der einzelnen Dateien, d.h. die Plazierung jeder Datei auf einer Disk. In dieser Tabelle sind auch Informationen über den noch verfügbaren Speicherplatz einer Disk abgelegt, in dem weitere Dateien erzeugt werden können.

Diese beiden Systembereiche einer Disk, die Verzeichnisse und die Dateizuordnungstabelle, ermöglichen es MS–DOS, die Dateien auf einer Disk zu organisieren und wiederzufinden.

Wenn Sie eine Disk neu formatieren, wird die Dateizuordnungstabelle neu eingerichtet und es wird ein leeres Verzeichnis erzeugt, das sogenannte Basis– oder Stamm–Verzeichnis (root directory).

4.1.3 Ein Verzeichnis anzeigen: Der DIR–Befehl

Sie können sich den Inhalt einer Disk mit DIR–Befehl anzeigen. Dieser Befehl weist MS–DOS an, alle Dateieinträge im aktuellen Vereichnis einer Disk in einem bestimmten Laufwerk anzuzeigen. Wenn Sie z.B. die erste MS–DOS-Systemdiskette im Laufwerk A (dem oberen Floppy Disk–Laufwerk) eingelegt haben und sich den Inhalt des aktuellen Verzeichnisses dieser Disk ansehen wollen, brauchen Sie nur einzugeben:

```
dir a:
```

Die Befehlsbestandteile im einzelnen:

dir Der Befehl selbst

a: Das Laufwerk A, in dem sich die Diskette befindet, deren aktuelles Verzeichnis angezeigt werden soll.

MS–DOS antwortet mit einer Verzeichnisliste, die alle Dateieinträge im aktuellen Verzeichnis Ihrer Systemdiskette enthält.

Jetzt sollte Ihr Bildschirm so aussehen:

```
Volume in drive A is MSDOS320-01
Directory of A:\

ANSI            SYS       1651      7.07.86    12.00
APPEND          COM       1725      7.07.86    12.00
ASSIGN          COM       1523      7.07.86    12.00
ATTRIB          EXE       8234      7.07.86    12.00
CHKDSK          EXE       9680      7.07.86    12.00
COMMAND         COM      23612      7.07.86    12.00
DISKCOMP        EXE       3808      7.07.86    12.00
DISKCOPY        EXE       4096      7.07.86    12.00
DRIVER          SYS       1102      7.07.86    12.00
EDLIN           EXE       7356      7.07.86    12.00
EXE2BIN         EXE       3050      7.07.86    12.00
FC              EXE      14558      7.07.86    12.00
FDISK           EXE      16830      7.07.86    12.00
FIND            EXE       6403      7.07.86    12.00
FORMAT          EXE      11005      7.07.86    12.00
GRAFTABL        EXE       8210      7.07.86    12.00
GRAPHICS        EXE      13170      7.07.86    12.00
JOIN            EXE       9012      7.07.86    12.00
KEYBDV          EXE       2886      7.07.86    12.00
KEYBGR          EXE       3141     26.11.86    12.05
KEYBSG          EXE       3149      8.11.86    19.45
KEYBSF          EXE       3149      8.11.86    19.45
KEYBIT          EXE       3138      8.11.86    11.58
KEYBUK          EXE       3045      7.11.86    23.35
LABEL           EXE       2750      7.07.86    12.00
MODE            EXE      13928      7.07.86    12.00
MORE            COM        282      7.07.86    12.00
PRINT           EXE       8824      7.07.86    12.00
RAMDRIVE        SYS       6462      7.07.86    12.00
RECOVER         EXE       4145      7.07.86    12.00
REPLACE         EXE       4852      7.07.86    12.00
SORT            EXE       1898      7.07.86    12.00
SUBST           EXE       9898      7.07.86    12.00
SYS             COM       4607      7.07.86    12.00
KEYBFR          EXE       3153      8.11.86    12.37
KEYBSP          EXE       3150      8.11.86    19.45
SETUP           COM       4603      3.12.86    18.56
AUTOEXEC        BAT        149      6.11.86    16.20
CONFIG          SYS         41      6.11.86    16.21
KEYBSV          EXE       3210     26.11.86    10.52
KEYBDA          EXE       3210     26.11.86    10.50
KEYBNO          EXE       3210     26.11.86    10.47
KEYBIC          EXE       3207     26.11.86    13.41
        43 File(s)                  46080 bytes free
```

Beachten Sie: Die Verzeichnisliste enthält **nicht** die Einträge für die beiden MS–DOS–Systemdateien IO.SYS und MSDOS.SYS, da diese verborgene Dateien sind, d.h. ihre Einträge beim DIR–Befehl nicht angezeigt werden.

Ebensogut können Sie sich auch Informationen über eine bestimmte Datei anzeigen lassen, indem Sie deren Namen beim DIR–Befehl explizit angeben. So erhalten Sie z.B. bei dem Befehl

```
dir report.txt
```

den Verzeichniseintrag für die Datei REPORT.TXT, der neben dem Namen auch die Dateigröße und Datum und Uhrzeit der Erzeugung oder letzten Fortschreibung enthält, angezeigt.

Weitere Informationen zum DIR–Befehl finden Sie im Kapitel 6.3.1 Allgemeine Befehle.

4.1.4 Eine Disk überprüfen: Der CHKDSK–Befehl

Der CHKDSK–Befehl führt bei der angegebenen Disk eine Datenkonsistenz– und Fehler–Prüfung durch, etwa ähnlich dem Probelesen eines gerade geschriebenen Briefes, ehe dieser abgeschickt wird. CHKDSK analysiert alle Verzeichnisse und die Dateizuordnungstabelle der angegebenen Disk und erzeugt anschließend einen Statusreport, in dem auch alle eventuell gefundenen Fehler angegeben werden. So werden z.B. Dateien angezeigt, die im Verzeichnis mit einer von Null verschiedenen Größe eingetragen sind, obwohl sie keine Daten enthalten.

Um z.B. die im Laufwerk A eingelegte Diskette zu überprüfen, brauchen Sie nur einzugeben:

```
chkdsk a:
```

Sie erhalten dann einen Statusreport und ggf. gefundene Fehler auf dem Bildschirm angezeigt. Wie diese Anzeige aussieht und zu interpretieren ist, wird im Kapitel 6.3.1 beim CHKDSK–Befehl beschrieben. Diesen Befehl sollten Sie in größeren Abständen bei allen wichtigen Disks anwenden, um die Integrität Ihrer Daten sicherzustellen.

4.1.5 Wie Dateien benannt werden

Der Name einer typischen MS–DOS–Datei hat das folgende Format:

```
handbuch.txt
```

Dateiname Dateinamenserweiterung

Der Name einer Datei besteht grundsätzlich aus zwei Teilen; im obigen Beispiel ist HANDBUCH der Dateiname und .TXT die Dateinamenserweiterung.

Der Dateiname darf 1 bis 8 Zeichen lang sein. Die Dateinamenserweiterung ist wahlfrei. Sie darf 1 bis 3 Zeichen lang sein und muß vom Dateinamen durch einen Punkt (.) getrennt sein. Ein Dateiname darf in Groß– oder Kleinbuchstaben oder einem Gemisch aus beiden angegeben werden. MS–DOS wandelt alle Buchstaben grundsätzlich intern in Großbuchstaben. Hier einige Beispiele für gültige Dateinamen:

```
RECHNUNG.MAI
BUDGET.86
MUELLER.BRF
Kapitel.Rep
adressen
```

Zusätzlich zum Dateinamen und der Dateinamenserweiterung kann der Dateiangabe eine Laufwerksbezeichnung hinzugefügt werden. Die Laufwerksbezeichnung weist MS–DOS an, die angegebene Datei auf der Disk in einem bestimmten Laufwerk zu suchen. Wenn z.B. der Verzeichniseintrag der Datei HANDBUCH.TXT, die auf der Disk in Laufwerk B gespeichert ist, angezeigt werden soll, wobei B **nicht** das aktuelle Laufwerk ist, müssen Sie den folgenden Befehl eingeben:

```
dir b:handbuch.txt
```

Sie erhalten dann Name, Größe sowie Datum und Uhrzeit der Erstellung oder letzten Fortschreibung der Datei auf dem Bildschirm angezeigt.

Wenn Laufwerk A das aktuelle Laufwerk (Standard–Laufwerk) ist, würde MS–DOS automatisch auf der Disk in diesem Laufwerk nach der Datei HANDBUCH.TXT suchen, wenn beim obigen DIR–Befehl kein Laufwerk angegeben worden wäre. Beim aktuellen Laufwerk ist also grundsätzlich keine Laufwerksbezeichnung erforderlich. Sie ist dagegegen immer dann

erforderlich, wenn sich ein eingegebener Befehl auf ein anderes als das aktuelle Laufwerk bezieht.

Neben Buchstaben dürfen Sie auch Ziffern sowie eine Reihe von Sonderzeichen in einem Dateinamen oder in der Dateinamenserweiterung verwenden. Hier die komplette Zusammenstellung aller erlaubten Zeichen bei Dateinamen:

A–Z a–z 0–9 $ &
% ' ( ) – @ ^
{ } ! ~ #

4.1.6 Verbotene Dateinamen

MS–DOS kennt selbst eine Reihe besonderer Dateinamen, die Sie deshalb für eigene Dateien **nicht** verwenden dürfen. Diese sind mit ihrer Bedeutung:

AUX: oder COM1:	Erster Anschluß für Datenfernübertragung
COM2:	Zweiter Anschluß für Datenfernübertragung
CON:	Tastatureingabe oder Bildschirmausgabe
PRN: oder LPT1:	Erster Paralleldrucker (nur Ausgabe)
LPT2:	Zweiter Paralleldrucker (nur Ausgabe)
LPT3:	Dritter Paralleldrucker (nur Ausgabe)
NUL:	Scheingerät, das nicht vorhanden ist, das aber dann angegeben werden kann, wenn ein Befehl keine bestimmte Datei erzeugen oder öffnen soll, wohl aber einen Ein– oder Ausgabedateinamen erfordert.

Auch wenn Sie diesen Dateinamen eine Dateinamenserweiterung anfügen, bleiben sie für eigene Dateien verboten. Z.B. ist CON.TXT ein verbotener Dateiname. Dagegen dürfen Sie diese oben angegebenen Namen als Dateinamenserweiterung ohne Einschränkung benutzen. Z.B. ist der Dateiname HANDBUCH.PRN erlaubt.

4.1.7 Jokerzeichen für globale Dateinamen

Zwei besondere Zeichen, hier als Jokerzeichen bezeichnet, können in Dateinamen verwendet werden, um eine größere Flexibilität bei der Dateinamensangabe in MS–DOS–Befehlen zu erzielen. Es sind dies der Stern (*) und das Fragezeichen (?).

4.1.7.1 Das ?–Jokerzeichen

Ein Fragezeichen (?) an einer beliebigen Stelle in einem Dateinamen oder einer Dateinamenserweiterung bedeutet, daß an dieser Stelle ein beliebiges erlaubtes Namenszeichen stehen darf. Z.B. erhalten Sie mit dem MS–DOS-Befehl

```
dir test?prg.exe
```

eine Liste aller Einträge im aktuellen Verzeichnis der Disk im aktuellen Laufwerk, deren Namen 8 Zeichen lang sind, mit den Buchstaben TEST beginnen, dann ein beliebiges erlaubtes Namenszeichen an 5. Position haben, mit den Buchstaben PRG enden und die Erweiterung .EXE haben, also z.B:

```
test1prg.exe
test2prg.exe
test3prg.exe
```

4.1.7.2 Das *–Jokerzeichen

Ein Stern (*) an einer beliebigen Position in einem Dateinamen oder einer Dateinamenserweiterung bedeutet, daß jedes oder jede beliebigen, erlaubten Namenszeichen diese und alle restlichen Positionen im Dateinamen besetzen können. Z.B. erhalten Sie mit dem MS–DOS–Befehl

```
dir test*.exe
```

eine Liste aller Einträge im aktuellen Verzeichnis der Disk im aktuellen Laufwerk, deren Namen mindestens 4 Zeichen lang sind, mit den Buchsta-

ben TEST beginnen, dann beliebige erlaubte weitere Namenszeichen ab der 5. Position und die Erweiterung .EXE haben, also z.B:

```
test.exe
test1prg.exe
testlauf.exe
```

Beachten Sie: Die Dateinamenabkürzung *.* bezieht sich auf **alle** Dateien in einem Verzeichnis und kann bei bestimmten MS–DOS–Befehlen zu sehr weitreichenden Befehlswirkungen führen. So löscht z.B. der Befehl

```
del *.*
```

alle Dateien im aktuellen Verzeichnis der Disk im aktuellen Laufwerk.

Beispiele

Wenn Sie die Einträge im aktuellen Verzeichnis der Disk im Laufwerk A für alle Dateien mit Namen RECHNUNG und beliebiger Dateinamenerweiterung sehen wollen, brauchen Sie nur einzugeben:

```
dir a:rechnung.*
```

Wenn Sie die Einträge im aktuellen Verzeichnis der Disk im Laufwerk B für alle Dateien mit bliebigem Namen und der Namenserweiterung TXT sehen wollen, brauchen Sie nur einzugeben:

```
dir b:*.txt
```

Das zweite Beispiel ist dann hilfreich, wenn Sie allen Ihren Textdateien die Namenserweiterung TXT geben und sich eine Liste aller Textdateien auf einer bestimmten Disk anzeigen lassen wollen, weil Sie den Namen einer bestimmten Textdatei vergessen haben.

4.1.8 Wie Dateien geschützt werden können

MS–DOS ist ein sehr leistungsfähiges und nützliches Hilfsmittel zur Verarbeitung persönlicher und geschäftlicher Informationen. Aber wie bei jedem Computer können auch hier Fehler auftreten oder Informationen können von Dritten mißbraucht werden. Wenn Sie mit Daten arbeiten, die unersetzlich sind oder einen hohen Grad an Geheimhaltung erfordern, sollten Sie

Maßnahmen ergreifen, die Ihre Programme vor Fremdbenutzung, –änderung oder –löschung schützen.

Einfache Maßnahmen wie Entfernen der Disketten, wenn Sie nicht mit dem Computer arbeiten, Anfertigen von Kopien wichtiger Disketten oder Verriegeln des Computers mit dem eingebauten Schloß gewährleisten bereits ein hohes Maß an Sicherheit für Ihre Daten und Programme.

Eine weitere Maßnahme ist die Schreibschutzaktivierung (Attribut R) mit dem ATTRIB–Befehl (s. dort in Kapitel 6.3).

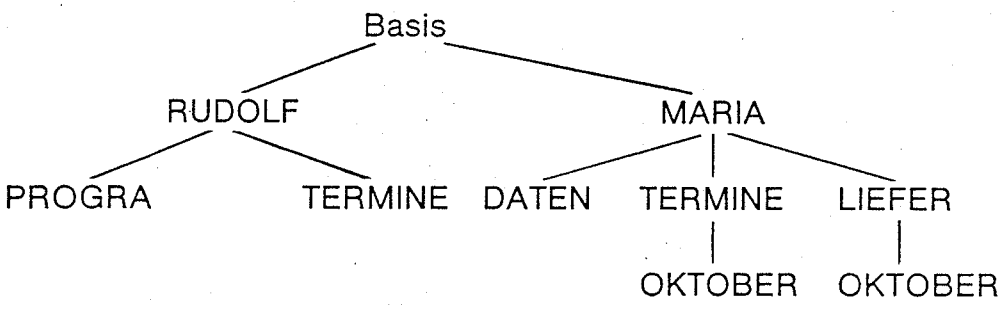
4.2 Disk–Verzeichnisse

Die Namen aller Dateien auf einer Disk werden in ein oder mehrere Verzeichnisse eingetragen. Zusätzlich enthalten diese Verzeichnisse Informationen über die Größe der Dateien sowie über Datum und Uhrzeit der Erzeugung oder letzten Änderung.

Wenn Ihr Computer noch von weiteren Anwendern benutzt wird, oder wenn Sie an verschiedenen Projekten arbeiten, können die Verzeichnisse durch immer neue Dateien so aufgebläht werden, daß sie unübersichtlich werden. Es erscheint deshalb sinnvoll, die Dateien nach Anwendern oder Arbeitsbereichen zu ordnen.

In einem Büro können Sie unterschiedliche Vorgänge in verschiedenen Aktenordnern ablegen. Ganz ähnlich erlaubt MS–DOS die Ablage von Dateien in unterschiedlichen Verzeichnissen. Verzeichnisse organisieren also Dateien in geeignete Datei–Gruppen. So kann es z.B. sinnvoll sein, alle Rechnungsdateien in einem Verzeichnis zusammenzufassen, alle Textdateien dagegen in einem anderen. Jedes Verzeichnis kann eine beliebige Anzahl Dateien enthalten und es kann weitere Verzeichnisse enthalten, die dann als Unterverzeichnisse bezeichnet werden. Diese Methode der Dateiorganisation wird auch als hierarchische Dateistruktur oder als baumstrukturierte Verzeichnisse bezeichnet.

Ein Beispiel einer solchen Struktur zeigt die untenstehende Abbildung:



Bei diesem Beispiel enthält das Basis–Verzeichnis der Diskette die beiden Verzeichnisse RUDOLF und MARIA. Das Verzeichnis RUDOLF enthält zwei Dateien namens PROGRA und TERMINE. Das Verzeichnis MARIA enthält eine Datei namens DATEN sowie zwei weitere Verzeichnisse namens TERMINE und LIEFER die beide je eine Datei namens OKTOBER enthalten.

Sie können sich eine hierarchische Dateistruktur als einen auf den Kopf gestellten Baum vorstellen, bei dem die Verzeichnisse die Äste und die Dateien die Blätter darstellen. Die Wurzel des Baumes stellt das sogenannte **Basis–Verzeichnis** dar. Dieses Verzeichnis wird automatisch angelegt, wenn eine Disk formatiert wird und auf ihr Dateien erzeugt werden. Sie können weitere Verzeichnisse und Unterverzeichnisse erzeugen, wie noch im weiteren Verlauf dieses Kapitels beschrieben werden wird.

Mit der Erzeugung weiterer Verzeichnisse für Dateigruppen oder andere Systembenutzer wächst die Dateistruktur. Innerhalb jedes neuen Verzeichnisses können sowohl neue Dateien als auch weitere Unterverzeichnisse erzeugt werden.

Sie können jederzeit durch die gesamte Dateistruktur "wandern" und jede Datei im System finden, wenn Sie beim Basis–Verzeichnis beginnen und dann den entsprechenden Ast des Verzeichnisbaumes entlangfahren, um zur gewünschten Datei zu gelangen. Umgekehrt können Sie auch von jeder beliebigen Postion im Verzeichnisbaum zurück zum Basis–Verzeichnis gelangen.

Das Verzeichnis, in dem Sie sich gerade befinden, wird **aktuelles Verzeichnis** genannt. Die im Kapitel 4.1.5 beschriebenen Dateinamen stehen in relativer Beziehung zum aktuellen Verzeichnis. D.h., wenn Sie Ihren Computer einschalten, befinden Sie sich im Basis–Verzeichnis, das gleichzeitig aktuelles Verzeichnis ist. Wenn Sie keine besonderen Maßnahmen ergreifen, wird jede neue Datei in diesem Verzeichnis erzeugt. Aus diesem Grunde können auf einer Disk mehrere Dateien desselben Namens existieren, wenn diese in unterschiedlichen Verzeichnissen abgelegt sind (s. obiges Beispiel).

Sie können im Basis–Verzeichnis Unterverzeichnisse mit Hilfe des MKDIR-Befehls (s. dort in Kapitel 6.3.1) einrichten. In dem o.a. Beispiel wurden vier Unterverzeichnisse erzeugt:

1. Ein persönliches Verzeichnis für den Benutzer RUDOLF.

2. Ein persönliches Verzeichnis für den Benutzer MARIA.

3. Zum Verzeichnis MARIA ein Unterverzeichnis TERMINE.

4. Zum Verzeichnis MARIA ein Unterverzeichnis LIEFER.

Die beiden Dateien OKTOBER stehen in keinerlei Beziehung zueinander, da sie in unterschiedlichen Verzeichnissen geführt werden.

Eine solche Verzeichnis– und Dateiorganisation ist solange für Sie als Benutzer unwesentlich, als Sie nur mit Dateien in Ihrem eigenen Verzeichnis arbeiten. Wenn Sie aber mit jemandem zusammenarbeiten oder zur selben Zeit an verschiedenen Projekten arbeiten, werden Sie die hierarchische Verzeichnisstruktur sehr hilfreich finden. Wenn Sie zum Beispiel eine Liste der Verzeichniseinträge in MARIAs Unterverzeichnis Lieferscheine haben wollen, brauchen Sie nur einzugeben:

```
dir \maria\liefer
```

Der nach links geneigte Schrägstrich (\) trennt dabei die einzelnen Verzeichnisse. An letzter Stelle einer solchen Angabe kann auch ein Dateinamen stehen. Der erste Schrägstrich trennt das danach stehende Verzeichnis vom Basis–Verzeichnis.

```
dir \rudolf
```

zeigt z.B. alle Verzeichniseinträge des Unterverzeichnis' RUDOLF. Dabei beginnt MS–DOS seine Suche nach diesem Verzeichnis beim Basis–Verzeichnis.

4.2.1 Verzeichnis–Pfade

Wenn Sie mit einem hierarchisch gegliederten Dateisystem arbeiten, müssen Sie MS–DOS angeben, wo gewünschte Dateien in diesem System angelegt sind oder werden sollen. Bei dem Beispiel im letzten Abschnitt hat z.B. die

Benutzerin MARIA sowohl im Unterverzeichnis LIEFER als auch im Unterverzeichnis TERMINE eine Datei namens OKTOBER. Wenn Sie mit der Datei OKTOBER im Verzeichnis TERMINE arbeiten will, muß sie eine spezielle Form der Dateiangabe verwenden, einen **Pfad–Namen**.

4.2.1.1 Pfad–Namen

Ein Pfadname ist Folge von Verzeichnisnamen mit oder ohne einem Dateinamen am Ende. In einem solchen Pfadnamen ist jeder Name vom vorhergehenden durch einen nach links geneigten Schrägstrich (\) getrennt. Die allgemeine Form eines Pfadnamens ist also:

[\][*Verzeichnis*]\[*Verzeichnis*]\. . .\[*Datei*]

Ein Pfadname darf beliebig viele Verzeichnisnamen enthalten, jedoch nur bis zu einer Maximallänge von 63 Zeichen. Beginnt er mit einem Schrägstrich, so beginnt MS–DOS die Suche nach dem Dateinamen im Basis–Verzeichnis, also ganz oben in der Verzeichnisstruktur. Andernfalls beginnt die Suche beim aktuellen Verzeichnis (Arbeitsverzeichnis). Die Pfadnamen für die beiden Dateien OKTOBER in MARIAs Unterverzeichnissen LIEFERSCHEINE und TERMINE im Beispiel des letzten Abschnittes lauten, wenn vom Basis-Verzeichnis ausgehend gesucht werden soll:

```
\maria\liefer\oktober
```
bzw.

```
\maria\termine\oktober.
```

Ein Verzeichnis, das Unterverzeichnisse enthält, wird als **Vater**–Verzeichnis bezeichnet. MS–DOS kennt zwei Abkürzungen für das aktuelle oder Arbeitsverzeichnis sowie das diesem übergeordnete Vater–Verzeichnis, die es grundsätzlich bei jedem Unterverzeichnis einträgt:

- **.** Ein Punkt anstelle des Dateinamens besagt, daß dieses Verzeichnis ein Unterverzeichnis ist. Diesen Eintrag macht MS–DOS automatisch, wenn es ein Unterverzeichnis anlegt.

- **..** Zwei Punkte anstelle des Dateinamens bezeichnen das diesem Unterverzeichnis übergeordnete, also das Vater–Verzeichnis. Die Angabe

  ```
  dir ..
  ```

listet die Einträge des Vater–Verzeichnisses zum aktuellen Verzeichnis.

```
dir ..\..
```

listet die Einträge des Vaterverzeichnisses zum Vaterverzeichnis zum aktuellen Verzeichnis.

4.2.1.2 Pfade und externe Befehle

Externe MS–DOS–Befehle (s.a. Kapitel 6.3.1) sind in Form von Programmen auf Disk gespeichert. Sie müssen vor ihrer Ausführung in den Speicher geladen werden.

Bei der Arbeit mit mehreren Verzeichnissen empfiehlt es sich, die externen MS–DOS–Befehle in einem separaten Verzeichnis zusammenzufassen.

Wenn Sie einen externen MS–DOS–Befehl benutzen, durchsucht MS–DOS zunächst das aktuelle Verzeichnis. Damit Sie nun nicht jedes mal den vollständigen Pfadnamen angeben müssen, können Sie MS–DOS mit Hilfe des PATH–Befehls (s. dort Kapitel 6.3.1) angeben, in welchem Verzeichnis nach externen Befehlen gesucht werden soll.

Wenn Sie sich z.B. im Unterverzeichnis \BIN\PROG befinden, und einen externen MS–DOS–Befehl aus dem Verzeichnis \BIN ausführen lassen wollen, so sollten Sie vorher den MS–DOS–Befehl

```
path \bin
```

eingeben. Dann durchsucht nämlich MS–DOS automatisch sowohl das aktuelle als auch das Verzeichnis BIN nach dem Befehl.

Den PATH–Befehl brauchen Sie nur ein einziges Mal während einer Arbeitssitzung mit Ihrem Computer zu geben. Wenn Sie diesen Befehl ohne Parameter benutzen, erhalten Sie den aktuell eingestellten Arbeitspfad angezeigt.

Beachten Sie: Der PATH–Befehl erlaubt MS–DOS, nur nach Programm–Dateien mit den Namenserweiterungen .BAT, .COM oder .EXE zu suchen.

4.2.1.3 Pfade und interne Befehle

Interne Befehle sind die einfachsten und am häufigsten benutzen MS–DOS-Befehle. Sie werden unverzüglich ausgeführt, da sie bereits in den Speicher geladen werden, wenn Sie MS–DOS starten, und dort resident verbleiben. Im nächsten Kapitel erfahren Sie nähere Einzelheiten zu den internen Befehlen.

Einige der internen Befehle können auch Pfade benutzen. So können Sie die Flexibilität der folgenden internen Befehle wesentlich erhöhen:

COPY ***Pfad Pfad***

Wenn der zweite Pfadname ein Verzeichnis ist, werden alle im ersten Pfadnamen angegebenen Dateien in dieses Verzeichnis kopiert. Beispielsweise werden mit dem folgenden Befehl alle Dateien im Unterverzeichnis LIEFER in das Unterverzeichnis RECHNUNG kopiert.

```
copy \maria\liefer \rechnung
```

DEL ***Pfad***

Bezeichnet der Pfadname ein Verzeichnis, so werden alle Dateien in diesem Verzeichnis gelöscht. MS–DOS fordert in diesem Fall mit der Anzeige

```
Are you sure (Y/N)?
```

(Sind Sie sicher (J/N)?) eine Bestätigung des globalen Löschbefehls vom Anwender. Bei **Y** (Yes = Ja) wird der Befehl ausgeführt, bei **N** (No = Nein) nicht. Auch hier ein Beispiel:

```
del maria\liefer
```

DIR ***Pfad***

Zeigt den Inhalt des in ***Pfad*** angegebenen Verzeichnisses. Ein Beispiel:

```
dir rudolf
```

TYPE *Pfad*

Bei diesem Befehl muß am Ende des Pfad–Namens ein Dateiname stehen. Der Inhalt dieser Datei wird dann auf dem Bildschirm angezeigt, also z.B.:

```
type \maria\termine\oktober
```

4.2.2 Die Arbeit mit Verzeichnissen

In den folgenden Abschnitten lernen Sie, wie das aktuelle oder Arbeitsverzeichnis angezeigt, erzeugt, gewechselt und gelöscht werden kann.

4.2.2.1 Das aktuelle Verzeichnis anzeigen

Das Verzeichnis, das aktiv ist, während ein MS–DOS–Befehl ausgeführt wird, wird als aktuelles Verzeichnis bezeichnet. Wenn Sie wissen wollen, welches Verzeichnis augenblicklich das aktuelle ist, brauchen Sie nur den CHDIR–Befehl **ohne** Pfadnamen einzugeben. Wenn Ihr aktuelles Verzeichnis z.B. \MARIA\TERMINE ist, so führt die Eingabe von

```
chdir
```

zu der Anzeige

```
a:\maria\termine
```

Es wird also sowohl das aktuelle Laufwerk (im obigen Beispiel A) als auch das aktuelle Verzeichnis angezeigt.

Hinweis: Zur Vereinfachung gibt es für den CHDIR–Befehl auch eine Abkürzung, nämlich CD. CD \MARIA\TERMINE ist dasselbe wie CHDIR \MARIA\TERMINE.

Wenn Sie den Inhalt des Verzeichnisses \MARIA\TERMINE sehen wollen, müssen Sie den DIR–Befehl verwenden. Sie erhalten dann eine Bildschirmanzeige der folgenden Art:

```
 Volume in drive A has no label
 Directory of  A:\MARIA\TERMINE

.                <DIR>       1.09.86   10.07
..               <DIR>      15.08.86    9.15
ALTTERM          <DIR)      15.08.86    8.26
SEPTEMB  TER       590       1.09.86   12.36
OKTOBER  TER       320       1.10.86   15.00
         2 File(s)          361686 bytes free
```

Dieser Diskette wurde kein Kennsatz zugewiesen, als sie formatiert wurde. Wie man einer Diskette einen Kennsatz zusweist, wird beim FORMAT– oder beim LABEL–Befehl in Kapitel 6.3.1 beschrieben. Beachten Sie bei der obigen Verzeichnisliste, daß sowohl Verzeichnisse als auch Dateien angezeigt werden. Das Verzeichnis \MARIA\TERMINE enthält außerdem ein weiteres Unterverzeichnis, nämlich ALTTERM. Die Angabe . bedeutet, daß das Verzeichnis \MARIA\TERMINE das aktuelle Verzeichnis ist, und .. bedeutet, daß es zu diesem Verzeichnis ein Vaterverzeichnis, nämlich \MARIA gibt. Die beiden Einträge SEPTEMB.TER und OKTOBER.TER bezeichnen Dateien. Alle diese Verzeichnisse und Dateien befinden sich auf der Disk in Laufwerk A.

Beachten Sie: Da Dateien und Verzeichnisse von MS–DOS bezüglich ihrer Einträge gleich behandelt werden, können Sie in einem Verzeichnis einer Datei nicht denselben Namen geben, wie einem Unterverzeichnis zu diesem Verzeichnis. Wenn Sie z.B. in dem Verzeichnis \MARIA sind, in dem es das Unterverzeichnis TERMINE gibt, so können Sie hier keine Datei mit dem Namen TERMINE erzeugen.

4.2.2.2 Ein Verzeichnis anlegen

Um im aktuellen Verzeichnis ein Unterverzeichnis anzulegen, können Sie den MKDIR–Befehl verwenden (s.a. dort in Kapitel 6.3.1). Wenn Sie zum aktuellen Verzeichnis z.B. ein Unterverzeichnis ALTTERM erzeugen wollen, brauchen Sie nur

```
mkdir altterm
```

einzugeben. Nach der Ausführung dieses Befehls existiert hierarchisch unterhalb des aktuellen Verzeichnisses ein neues Verzeichnis. Wenn Sie anstelle des neuen Verzeichnisnamens beim MKDIR–Befehl einen Pfad angeben, können Sie an beliebiger Stelle Ihrer Verzeichnisstruktur ein neues Verzeichnis anlegen. Beim neuen Verzeichnis erzeugt MS–DOS außerdem automatisch die beiden .– und . .–Einträge.

Beachten Sie: Die Länge eines Pfadnamens ist auf 63 Zeichen begrenzt!

Um nun Dateien in dem neuen Verzeichnis zu erzeugen, können Sie z.B. den Zeileneditor EDLIN (s. Kapitel 8) benutzen, oder einfach andere Dateien aus anderen Verzeichnissen dort anlegen.

4.2.2.3 Das aktuelle Verzeichnis wechseln

Um vom aktuellen Verzeichnis aus ein anderes Verzeichnis zu aktualisieren, dient der bereits oben erwähnte CHDIR–Befehl zusammen mit einem Pfadnamen. Wenn Sie z.B.

```
chdir \maria
```

eingeben, wird das Verzeichnis MARIA zum aktuellen Verzeichnis.

Sie können jeden beliebigen Pfadnamen angeben, um in Ihrer Verzeichnisstruktur herumzuwandern. Wenn Sie

```
chdir ..
```

eingeben, wird das Vaterverzeichnis des aktuellen Verzeichnisses zum neuen, aktuellen Verzeichnis.

4.2.2.4 Ein Verzeichnis löschen

Um ein Verzeichnis aus Ihrer Verzeichnisstruktur zu löschen, verwenden Sie den RMDIR–Befehl (s.a. dort in Kapitel 6.3.1). Um z.B. das Verzeichnis ALTTERM aus dem aktuellen Verzeichnis zu löschen, brauchen Sie nur

```
rmdir altterm
```

einzugeben. Dabei muß jedoch dieses Verzeichnis bis auf die beiden .– und . .–Einträge leer sein. Damit wird verhindert, daß aus Versehen vielleicht wichtige Dateien mitgelöscht werden, wenn ein Verzeichnis gelöscht wird. Wenn beim RMDIR–Befehl ein Pfadname angegeben wird, kann jedes beliebige Verzeichnis aus jedem beliebigen anderen Verzeichnis in der Struktur heraus gelöscht werden. Um z.B. vom Verzeichnis RUDOLF aus das Verzeichnis TERMINE im Verzeichnis MARIA zu löschen, brauchen Sie sich nur vorher zu vergewissern, daß das Verzeichnis TERMINE bis auf die beiden .– und . .–Einträge leer ist und dann einzugeben:

```
rmdir \maria\termine
```

Um alle Dateien bis auf die beiden .– und . .–Einträge in einem Verzeichnis zu löschen, benutzen Sie den DEL–Befehl zusammen mit dem Pfadnamen des entsprechenden Verzeichnisses. Um z.B. alle Dateien im Verzeichnis TERMINE zu löschen, geben Sie einfach ein:

```
del \maria\termine
```

Wie oben bereits erwähnt, fordert MS–DOS mit der Anzeige

```
Are you sure (Y/N)?
```

(Sind Sie sicher (J/N)?) eine Bestätigung. Wenn Sie N eingeben, wird der Befehl nicht ausgeführt.

Die beiden .– und . .–Einträge können mit dem DEL–Befehl nicht gelöscht werden. Sie werden von MS–DOS als Teil der hierarchischen Verzeichnisstruktur erzeugt.

4.2.2.5 Ein Verzeichnis umbenennen

MS–DOS kennt keinen Befehl zur Umbenennung eines Verzeichnisses. Sie können jedoch ein Verzeichnis, das keine weiteren Unterverzeichniseinträge enthält, umbenennen, wenn Sie diesen Schritten folgen:

1. Erzeugen Sie ein neues Verzeichnis mit dem neuen Namen mit Hilfe des MKDIR–Befehls (s.a. dort in Kapitel 6.3.1).

2. Kopieren Sie mit Hilfe des COPY–Befehls (s. dort in Kapitel 6.3.1) alle Dateien aus dem alten Verzeichnis in das neue Verzeichnis.

3. Löschen Sie alle Dateien im alten Verzeichnis mit dem DEL–Befehl (s. dort in Kapitel 6.3.1).

4. Löschen Sie das alte Verzeichnis mit dem RMDIR–Befehl (s. dort in Kapitel 6.3.1).

Dazu ein Beispiel:

Um das Verzeichnis \MARIA\TERMINE in \MARIA\KALENDER umzubenennen, geben Sie nacheinander folgende 4 Befehle ein und drücken Sie nach jedem Befehl die Eingabetaste:

1. `mkdir \maria\kalender`

2. `copy \maria\termine \maria\kalender`

3. `del \maria\termine`

4. `rmdir \maria\termine`

Sie haben in diesem Kapitel die Anwendung der folgenden internen MS–DOS–Befehle gelernt:

- DIR zeigt alle Einträge in einem Verzeichnis an.
- CHKDSK überprüft eine Disk.
- PATH setzt einen MS–DOS–Suchpfad.
- CHDIR zeigt den Namen des aktuellen Verzeichnisses an oder wechselt das aktuelle Verzeichnis.
- MKDIR legt ein neues Verzeichnis an.
- RMDIR löscht ein leeres Verzeichnis.

Im nächsten Kapitel erfahren Sie Einzelheiten zu den allgemeinen Eigenschaften der MS–DOS–Befehle.

5. Allgemeines über Befehle

5.1 Was ist ein Befehl?

Ein Befehl ist der Weg, über den Sie als Anwender mit Ihrem Computer kommunizieren. Durch die Eingabe von Befehlen über die Tastatur des Computers weisen Sie diesen an, die gewünschte Arbeit für Sie zu erledigen. MS–DOS stellt Befehle für die folgenden Aufgaben zur Verfügung:

- Vergleichen, Kopieren, Anzeigen, Löschen und Umbenennen von Dateien;
- Formatieren und Duplizieren ganzer Disketten oder Festplatten;
- Ausführen von kompletten MS–DOS–Programmen (z.B. EDLIN oder DEBUG) oder Anwenderprogrammen;
- Analysieren und Auflisten von Dateivezeichnissen;
- Einstellen von Datum und Uhrzeit;
- Einrichten und Ausführen von Stapelverarbeitungsdateien;
- Setzen von verschiedenen Drucker– und Bildschirm–Betriebsarten;
- Setzen von Programmunterbrechungen für bestimmte Zeiten.

5.2 Befehlsarten bei MS–DOS

MS–DOS unterscheidet zwei Befehlsarten:

- Interne Befehle
- Externe Befehle

Interne Befehle sind einfache und häufig benutzte Befehle, die nicht in separaten Dateien auf der Systemdiskette gespeichert sind und deshalb auch nicht in deren Verzeichnis auftauchen. Sie sind vielmehr Bestandteil der Datei COMMAND.COM. Interne Befehle werden unmittelbar nach ihrer Eingabe ausgeführt, da sie beim Starten von MS–DOS zusammen mit der Datei COMMAND.COM in den Speicher des Computers geladen werden, und dort solange resident verbleiben, bis der Computer wieder ausgeschaltet wird. Folgende Befehle, die im einzelnen im Kapitel 6.3.1 beschrieben werden, gehören dazu:

BREAK	DIR	PATH	SHIFT
CHDIR	ECHO	PAUSE	TIME
CLS	EXIT	PROMPT	TYPE
COPY	FOR	REM	VER
CTTY	GOTO	REN	VERIFY
DATE	IF	RMDIR	VOL
DEL	MKDIR	SET	

Externe Befehle sind in Form von Programmdateien auf der Systemdisk gespeichert. Sie müssen von der Disk in den Hauptspeicher des Computers geladen werden, ehe sie ausgeführt werden können.

MS–DOS betrachtet jede Programmdatei mit der Namenserweiterung .BAT, .COM oder .EXE als externen Befehl. So sind z.B. die Programmdateien FORMAT.COM oder DISKCOPY.COM externe Befehle. Da alle externen Befehle Dateien sind, können Sie sich auch Ihre eigenen Befehle programmieren und den externen MS–DOS–Befehlen hinzufügen. Eigene Programme, die Sie mit den üblichen Programmiersprachen einschließlich des Assemblers erzeugen, erhalten bei MS–DOS die Namenserweiterung .EXE.

Wenn Sie einen externen Befehl eingeben, brauchen Sie die Dateinamenserweiterung (.COM oder .EXE) nicht mit anzugeben.

Folgende Befehle, die im einzelnen im Kapitel 6.3.1 beschrieben werden, gehören dazu:

APPEND	DISKCOPY	JOIN	RESTORE
ASSIGN	EXE2BIN	LABEL	SHARE
ATTRIB	FDISK	MODE	SORT
BACKUP	FIND	MORE	SUBST
CHKDSK	FORMAT	PRINT	SYS
COMMAND	GRAFTABL	RECOVER	TREE
DISCOMP	GRAPHICS	REPLACE	XCOPY

5.3 Befehlsparameter

Mit Hilfe von wählbaren Parametern können Sie die Bedeutung der MS–DOS–Befehle modifizieren. Wenn Sie bei einem Befehl keinen Parameter angeben, so setzt MS–DOS dafür den sogenannten voreingestellten Parameter. Bei den einzelnen Befehlsbeschreibungen in Kapitel 6.3.1 sind bei jedem Befehl die voreingestellten Parameter angegeben.

Ein in eckige Klammern ([ ]) eingefaßter Parameter bei einer Befehlsbeschreibung bedeutet, daß dieser Parameter wahlfrei ist, d.h. er kann, muß aber nicht angegeben werden. Alle Angaben in *kursiver* Schrift bedeuten, daß dafür der angegebene Parameter in der von MS–DOS vorgeschriebenen Schreibweise eingesetzt werden muß. Beachten Sie außerdem, daß eine Laufwerksbezeichnung nur dann angegeben werden muß, wenn MS–DOS zur Befehlsausführung ein anderes, als das aktuelle Laufwerk verwenden soll.

Es gilt also folgendes ganz allgemeines Format für alle MS–DOS–Befehle:

Befehl [***Parameter***]

wobei folgende ***Parameter*** vorkommen können:

Laufwerk	bezieht sich auf eine Laufwerksbezeichnung nach MS–DOS–Konventionen.
Dateiname	bezieht sich auf einen beliebigen Dateinamen einschließlich der ggf. vorhandenen Dateinamenserweiterung nach MS–DOS–Konventionen. Sie bezieht sich jedoch **nicht** auf eine Geräte– oder Laufwerksbezeichnung.
Pfadname	bezieht sich auf einen Pfad– oder Dateinamen in dem folgenden allgemeinen Format: [*Verzeichn*]\[*Verzeichn*]\. . .\[*Datei*]
Schalter	bezieht sich auf Auswahlmöglichkeiten, die die Wirkung eines bestimmten Befehls steuern. Ein Schalter ist ein Buchstabe, dem ein Schrägstrich (/) vorangestellt ist (z.B. /P).
Argumente	bezieht sich auf wählbare Alternativen eines Parameters, z.B. ON und OFF.

5.4 Was Sie bei der Befehlseingabe wissen sollten

Folgende Regeln sollten Sie bei der Benutzung von MS–DOS–Befehlen berücksichtigen:

1. Auf Befehle folgen gewöhnlich ein oder mehrere Parameter, wie z.B. Dateinamen.

2. Befehle und Parameter können in Groß–, in Kleinbuchstaben oder in gemischter Schreibweise eingegeben werden. MS–DOS wandelt alle Buchstaben intern grundsätzlich in Großschrift um.

3. Befehle und Parameter müssen durch bestimmte Zeichen oder Leerstellen voneinander getrennt werden. Gwöhnlich sind dies das Leerzeichen oder das Komma. Dafür zwei Beispiele:

   ```
   del neudatei.neu neudatei.txt
   rename,report.txt brief.txt
   ```

   Als weitere Trennzeichen sind das Semikolon (;), das Gleichheitszeichen (=) oder die Tabulatortaste erlaubt.

   Im weiteren Verlauf dieses Handbuches wird als Trennzeichen das Leerzeichen verwendet, also z.B.

   ```
   copy notiz1.txt notiz2.txt
   ```

4. Befehle werden erst nach dem Drücken der Eingabe–Taste ausgeführt.

5. Wird der Anwender bei einem Befehl mit folgender Meldung konfrontiert

   ```
   Strike a key when ready...
   ```

   (Wenn bereit, eine Taste drücken), so wird der Befehl ausgeführt, wenn eine beliebige Buchstaben– oder Zifferntaste oder die Leertaste gedrückt wird.

6. Ist bei einem Befehl die Angabe eines Dateinamens erforderlich, so muß die Namenserweiterung mit angegeben werden, wenn sie existiert.

7. Die Befehlsausführung kann durch gleichzeitiges Drücken der Tasten Ctrl und Break oder Ctrl und C abgebrochen werden.

8. Wenn ein Befehl große Datenmengen auf dem Bildschirm ausgibt, wird der Schirm automatisch aufgerollt, wenn die Anzeige die unterste Zeile ausgefüllt hat. Dieses Aufrollen kann durch gleichzeitiges Drücken der Tasten Ctrl und NumLock oder Ctrl und S angehalten werden. Jede beliebige andere Taste setzt die Bildschirmausgabe fort.

9. Die eingebene Befehlszeile kann mit Hilfe der MS–DOS–Funktions– und –Ediertasten bearbeitet werden. Im Kapitel 7 sind Einzelheiten dazu beschrieben.

10. Disk–Laufwerke werden in diesem Handbuch häufig als Quell– und Ziellaufwerk bezeichnet. Das Quellaufwerk ist das Laufwerk, das die Diskette enthält, **von** der Informationen übertragen werden sollen. Das Ziellaufwerk ist das Laufwerk, das die Diskette enthält, **auf** die Informationen übertragen werden sollen.

5.5 Ein– und Ausgabe

MS–DOS erwartet bei der Befehlseingabe grundsätzlich die Daten von der Tastatur und gibt Daten auf dem Bildschirm aus. Sie können jedoch den Strom der Befehlsein– und/oder –ausgabe auch umleiten. So kann die Eingabe aus einer Datei anstatt von der Tastatur kommen und die Ausgabe kann anstatt auf dem Bildschirm in eine Datei oder auf einem Drucker erfolgen. Zusätzlich haben Sie die Möglichkeit, durch Verwendung sogenannter Filter–Befehle die Ausgabe eines Befehls als Eingabe eines anderen Befehls zu verwenden.

In den nächsten Abschnitten lernen Sie, wie man die Befehls–Ein– und –Ausgabe umleitet und wie man Filter effektiv einsetzt.

5.5.1 Umleiten der Standard–Ein– und –Ausgabe

Die meisten Befehle senden ihre Ausgabe an den Bildschirm des Computers. Diese Ausgabe können Sie in eine Datei umleiten, indem Sie das Größer-Symbol (>) im Befehl verwenden. Z.B. liefert der Befehl

```
dir
```

eine Liste aller Verzeichniseinträge der Disk im aktuellen Laufwerk und zeigt

diese Liste auf dem Bildschirm an. Diese Liste wird jedoch in die Datei VERZEICH auf der Disk im aktuellen Laufwerk ausgegeben, wenn Sie z.B. den Befehl

```
dir >verzeich
```

verwenden. Wenn die Datei VERZEICH noch nicht existiert, wird sie von MS–DOS im aktuellen Verzeichnis eingerichtet und die Daten werden dort gespeichert. Ist sie bereits vorhanden, werden die alten Daten von den neuen überschrieben.

Wenn Sie jedoch die Ausgabe eines Befehls an das Ende einer bereits existierenden Datei anfügen wollen, können Sie dies z.B. mit dem Befehl

```
dir >>verzeich
```

erreichen. Zwei Größer–Symbole hintereinander bewirken also ein Anfügen der Befehlsausgabe an eine existierende Datei. Auch hier wird die Datei VERZEICH neu eingerichtet, wenn sie noch nicht existiert.

Bei der Arbeit mit MS–DOS ist es häufig hilfreich, die Eingabe einem Befehl nicht von der Tastatur, sondern aus einer Datei zuzuführen. Eine solche Umleitung können Sie durch die Angabe des Kleiner–Symbols (<) im Befehl erreichen. So erwartet z.B. der Befehl

```
sort <namen >list1
```

die Eingabedaten aus der Datei NAMEN, sortiert diese und gibt sie in die Datei LIST1 aus. Beide Dateien befinden sich in diesem Fall im aktuellen Verzeichnis der Disk im aktuellen Laufwerk (nähere Einzelheiten zum SORT–Befehl entnehmen Sie bitte dem Kapitel 6.3.1).

5.5.2 Filter

Ein Filter ist ein Befehl, der die Eingabe liest, sie auf bestimmte Weise verarbeitet und dann üblicherweise auf dem Bildschirm oder in eine Datei ausgibt. Die Daten werden also durch einen solchen Befehl gewissermaßen gefiltert. Da Filter auf verschiedene Art und Weise kombiniert werden können, können wenige Filter eine große Anzahl spezieller Befehle ersetzen.

Zu den MS–DOS–Filtern gehören die folgenden Befehle:

FIND	durchsucht eine angegebene Datei nach einer bestimmten Zeichenkette.
MORE	zeigt Text bildschirmweise an und wartet auf Quittierung.
SORT	sortiert Textdateien zeilen– oder satzweise.

Wie diese Filter verwendet werden, lernen Sie im nächsten Abschnitt.

5.5.2.1 Befehlsverkettung mit Filtern

Wenn Sie MS–DOS mehrere Befehle auf einmal geben wollen, können Sie diese miteinander verketten. So kann es z.B. erforderlich sein, die Ausgabe eines Befehls einem anderen Befehl als Eingabe direkt zuzuleiten. Ein typisches Beispiel dafür ist ein Programm, das seine Ausgabe spaltenweise erzeugt und diese Spalten sollen sortiert werden.

Die Verkettung von MS–DOS–Befehlen unter Verwendung von Filtern geschieht mit dem senkrechten Strich (|). So erzeugt z.B. der Befehl

```
dir | sort
```

ein sortiertes Dateiverzeichnis der Disk im aktuellen Laufwerk und zeigt dieses auf dem Bildschirm an. Der senkrechte Strich (|) bewirkt, daß die gesamte Ausgabe des links von ihm stehen Befehls dem rechts von ihm stehenden Befehl zur Weiterverarbeitung zugeführt wird.

Die Befehlsverkettung kann auch mit der Umleitung der Standard–Ein–/Ausgabe kombiniert werden. Der Befehl

```
dir | sort >verzeich.srt
```

erzeugt eine Datei VERZEICH.SRT im aktuellen Verzeichnis der Disk im aktuellen Laufwerk, in der das sortierte Verzeichnis dieser Disk abgelegt wird. Selbstverständlich können Sie dem Dateinamen auch eine Laufwerksbezeichnung voranstellen, wenn die Ausgabe auf eine andere als die aktuelle Disk erfolgen soll, also z.B.:

```
dir | sort >b:verzeich.srt
```

Natürlich können auch mehr als zwei Befehle miteinander verkettet werden:

```
dir | sort | more
```

Bei dieser Befehlskette wird das Verzeichnis der Disk im aktuellen Laufwerk zunächst sortiert und dann bildschirmweise angezeigt. Jedesmal, wenn ein Bildschirm vollgeschrieben ist, erzeugt der MORE–Filter die Anzeige

```
-- More --
```

Die Ausgabe wird dann fortgesetzt, wenn eine beliebige Taste gedrückt wird.

Je vertrauter Sie mit MS–DOS werden, umso mehr Anwendungen werden Sie für die Filter–Befehle und die Befehlsverkettung finden.

5.6 Stapelverarbeitung

Sie werden bei der Arbeit mit MS–DOS sehr bald feststellen, daß Sie zur Ausführung bestimmter Arbeiten dieselben Befehlsfolgen immer und immer wieder eingeben. Um sich diese lästigen Wiederholungen zu ersparen, erlaubt MS–DOS die Erstellung sogenannter Befehlsdateien. Diese Dateien sind reine Textdateien, in denen die nacheinander auszuführenden Befehle zeilenweise abgelegt sind. Solche Dateien nennt man Stapeldateien. Sie werden einfach durch Eingabe ihres Namens aufgerufen und die in ihnen abgelegten Befehle werden in der vorgegebenen Reihenfolge genauso abgearbeitet, als wenn sie über die Tastatur eingegeben worden wären.

Stapeldateien müssen die Namenserweiterung .BAT erhalten und werden durch Eingabe ihres Namens **ohne** die Erweiterung aufgerufen.

Eine solche Stapeldatei kann mit Hilfe des Zeileneditors EDLIN (s.a. Kapitel 8) oder einer speziellen Form des COPY–Befehls erzeugt werden. Wie dies bewerkstelligt wird, wird in den folgenden Abschnitten ausführlich beschrieben.

Zwei typische Stapelverarbeitungsbefehle, also Befehle die nur in Stapelverarbeitungsdateien verwendet werden dürfen, sind die Befehle REM und PAUSE, die detailliert im Kapitel 6.3.2 beschrieben werden. Mit REM können Kommentare in Stapelverarbeitungsdateien eingefügt werden. Mit PAUSE können wahlweise Benutzernachrichten angezeigt werden und die Stapel-

verarbeitung kann an einem definierten Punkt unterbrochen oder fortgeführt werden.

Stapelverarbeitung wird immer dann eingesetzt, wenn mehrere MS–DOS-Befehle nacheinander durch einen einzigen Befehl ausgeführt werden sollen. Z.B. sieht die Stapelverarbeitung für das Formatieren und Überprüfen einer neuen Diskette so aus:

```
rem Dies ist eine Stapelverarbeitungsdatei zur
rem Überprüfung neuer Disketten
rem Sie heißt NEUDISK.BAT
pause Bitte neue Diskette in Laufwerk A: einlegen
format a:
chkdsk a:
```

Um diese Stapelverabeitungsdatei auszuführen, nachdem sie z.B. mit EDLIN erzeugt und unter dem Namen NEUDISK.BAT gespeichert wurde, wird nur der Dateiname ohne die Namenserweiterung eingegeben, also:

```
neudisk
```

Das Ergebnis ist dasselbe, als wenn die letzten beiden Zeilen der Stapelverarbeitungsdatei nacheinander als Befehle eingegeben und ausgeführt worden wären.

Hier noch einmal die Schritte, die zur Erzeugung und Ausführung einer Stapeldatei erforderlich sind.

1. Aufruf des Zeileneditors EDLIN (s.a. Kapitel 8).

2. Eingabe der einzelnen auszuführenden MS–DOS–Befehle, jeder für sich in einer eigenen Zeile.

3. Speichern der Zeilen in einer Datei mit der Namenserweiterung .BAT.

4. Aufruf der Stapeldatei durch Eingeben des Dateinamens ohne die Erweiterung .BAT.

Ehe Sie jedoch eine Stapelverarbeitung ablaufen lassen, sollten Sie die nachfolgend aufgeführten Punkte berücksichtigen:

1. Zum Aufruf einer Stapelverarbeitungsdatei wird nur ihr Name, **nicht** die Namenserweiterung angegeben.

2. Wenn Sie während einer Stapelverarbeitung die Tastenkombination Ctrl–Break oder Ctrl–C drücken, erhalten Sie die Anzeige

   ```
   Terminate batch job (Y/N)?
   ```

   (Stapeljob abbrechen (J/N)?). Antworten Sie hier mit Y (Yes = Ja), so werden die restlichen Befehle in der Stapelverarbeitungsdatei nicht mehr ausgeführt und MS–DOS kehrt mit seiner Bereitschaftsanzeige wieder auf die Befehlsebene zurück. Wird mit N (No = Nein) geantwortet, so wird der aktuelle Befehl abgebrochen und die Stapelverarbeitung mit dem nächsten Befehl aus der Datei fortgesetzt.

3. Wenn Sie während der Stapelverarbeitung die Diskette mit der Stapelverarbeitungsdatei aus dem Laufwerk nehmen, fordert MS–DOS Sie auf, die Diskette wieder einzulegen, ehe der nächste Befehl ausgeführt wird.

4. Der letzte Befehl in einer Stapelverarbeitungsdatei kann der Name einer weiteren Stapelverarbeitungsdatei sein. Auf diese Weise lassen sich mehrere Stapeldateien miteineinander verknüpfen, um auch komplexeste Befehlsfolgen ausführen zu lassen.

5. Auch in einer Stapelverarbeitungsdatei können Standard–Ein– und –Ausgabe mit Hilfe der Symbole < und > umgeleitet werden. Die Befehlsverkettung von Filtern mit dem |–Symbol (s.a. Kapitel 5.5.2.1) ist dagegen bei Stapelverarbeitungsdateien nicht möglich.

5.6.1 Die AUTOEXEC.BAT–Datei

MS–DOS sucht beim Start nach einer Stapelverarbeitungsdatei mit dem Namen AUTOEXEC.BAT und führt die darin abgelegten Befehle automatisch aus, wenn die Datei vorhanden ist. Dies ist dann hilfreich, wenn mit dem Einschalten des Computers auch gleich ein bestimmtes Programm gestartet werden soll. Wenn eine AUTOEXEC.BAT–Datei erzeugt werden soll, so muß sie im Basisverzeichnis der Disk, von der aus MS–DOS gestartet wird, angelegt werden.

Die AUTOEXEC.BAT–Datei wird, wenn sie vorhanden ist, sofort ausgeführt und die sonst übliche Datums– und Zeit–Anforderung wird unterdrückt, es sei denn, die Befehle dafür sind in der AUTOEXEC.BAT–Datei vorhanden.

5.6.2 Wie eine AUTOEXEC.BAT–Datei erzeugt wird

Wenn Sie z.B. jedesmal, wenn Sie MS–DOS starten, den GW BASIC–Interpreter laden und ein BASIC–Programm namens MENUE aufrufen wollen, so können Sie die dafür erforderliche AUTOEXEC.BAT–Datei folgendermaßen erzeugen:

1. Geben Sie ein:

   ```
   copy con: autoexec.bat
   ```

   Dieser Befehl weist MS–DOS an, die über die Tastatur (Gerät CON) eingegebenen Informationen in eine Datei namens AUTOEXEC.BAT auf der Disk im aktuellen Laufwerk zu kopieren. Beachten Sie dabei, daß das Basis–Verzeichnis dieser Disk das aktuelle Verzeichnis sein **muß**.

2. Geben Sie jetzt ein:

   ```
   gwbasic menue
   ```

   und drücken Sie zum Abschluß die Eingabetaste. Dieser Befehl wird jetzt in die AUTOEXEC.BAT–Datei geschrieben. Er weist MS–DOS an, nach dem Start den GW BASIC–Interpreter zu laden und dann das GW BASIC–Programm MENUE aufzurufen und zu starten.

3. Drücken Sie jetzt die Tastenkombination Ctrl–Z und anschließend die Eingabetaste, um die Dateneingabe in die AUTOEXEC.BAT–Datei abzuschließen.

Sie können jede beliebige MS–DOS–Befehlssequenz in die AUTOEXEC-.BAT–Datei schreiben.

Hinweis: Wie bereits erwähnt, übergeht MS–DOS die Datums– und Zeitanforderung, wenn eine AUTOEXEC.BAT–Datei vorhanden ist. Es empfiehlt sich jedoch dringend, diese beiden Befehle in die AUTOEXEC.BAT–Datei aufzunehmen, da MS–DOS bei der Fortschreibung der Diskverzeichnisse sowohl Datum als auch Uhrzeit vermerkt und so dem Anwender die Möglichkeit bietet, die Aktualität seines Dateisystems zu überprüfen. Die Befehle DATE und TIME sind detailliert im Kapitel 6.3.1 beschrieben. Besitzen Sie einen PC 40, so benötigen Sie beide Befehle nicht, da dieser Computer über eine batteriegepufferte Systemuhr verfügt.

5.6.3 Eine Stapeldatei mit austauschbaren Parametern erzeugen

Es kann der Fall eintreten, daß ein Programm wiederholt mit unterschiedlichen Daten, die in verschiedenen Dateien abgelegt sind, abgearbeitet werden soll.

Ein Parameter in einem MS–DOS–Befehl stellt ja eine Option dar, die der Anwender definiert. MS–DOS bietet deshalb die Möglichkeit, in einer Stapelverarbeitungsdatei austauschbare Parameter anzugeben, die dann durch echte Werte ersetzt werden, wenn die Stapelverarbeitungsdatei ausgeführt wird. Zehn solcher Parameter können in einer Stapelverarbeitungsdatei verwendet werden. Sie werden mit %0 bis %9 bezeichnet.

Geben Sie z.B. ein:

```
copy con: neudatei.bat
del %3.doc
copy %1.doc+%2.doc %3.doc
print %3.doc
```

Drücken Sie dann Ctrl–Z und anschließend die Eingabetaste. MS–DOS antwortet dann mit

```
1 File(s) copied
A>_
```

(1 Datei(en) kopiert). Damit wurde die Datei NEUDATEI.BAT, in der drei Befehle enthalten sind, auf der Disk im aktuellen Laufwerk erzeugt.

Die austauschbaren Parameter %1 und %2 werden sequentiell durch die Parameter ersetzt, die Sie beim Aufruf der Stapelverarbeitungsdatei hinter deren Namen angeben. Der Parameter %0 wird generell durch den Laufwerksnamen (falls angegeben) und durch den Namen der Stapelverarbeitungsdatei (NEUDATEI.BAT im obigen Beispiel) ersetzt.

Beachten Sie:

1. Grundsätzlich können Sie bis zu 10 austauschbare Parameter in einer Stapelverarbeitungsdatei verwenden. Falls Sie mehr benötigen, müssen Sie den SHIFT–Befehl (s. dort in Kapitel 6.3.2) verwenden.

2. Wenn Sie innerhalb einer Stapelverarbeitungsdatei einen Dateinamen verwenden wollen, der selbst ein Prozentzeichen (%) enthält, so muß hier

das Prozentzeichen doppelt angegeben werden. Eine Datei namens ABC%.EXE muß also, wenn sie in einer Stapelverarbeitungsdateiangegeben wird, als ABC%%.EXE definiert werden.

5.6.4 Eine Stapeldatei abarbeiten

Um die im letzten Abschnitt erzeugte Stapelverarbeitungsdatei NEUDATEI.BAT abzuarbeiten und dabei die Parameter anzugeben, die die austauschbaren Parameter in der Datei ersetzen sollen, müssen Sie den Namen der Stapelverarbeitungsdatei (ohne die Namenserweiterung .BAT) und dahinter die Parameter, die die Parameter %1, %2 und %3 ersetzen sollen, eingeben. Die Stapelverarbeitungsdatei NEUDATEI.BAT enthält die drei Befehlszeilen

```
del %3.doc
copy %1.doc + %2.doc %3.doc
print %3.doc
```

Sie müssen also zum Ausführen der Stapelverarbeitungsdatei z.B. eingeben:

```
neudatei a:notiz1 b:notiz2 b:report
```

und dann die Eingabetaste drücken. A:NOTIZ1 ersetzt dann %1, B:NOTIZ2 ersetzt %2 und B:REPORT wird für 3% gesetzt.

Das Ergebnis ist dasselbe, als wenn Sie nacheinander die drei Befehle

```
del b:report.doc
copy a:notiz1.doc + b:notiz2.doc b:report.doc
print b:report.doc
```

eingegeben hätten.

Die folgende Tabelle zeigt, wie MS–DOS bei dem obigen Beispiel jeden der drei Parameter in der Stapeldatei ersetzt:

Stapeldatei:	Parameter 1 %1	Parameter 2 %2	Parameter 3 %3
NEUDATEI:	A:NOTIZ1.DOC	B:NOTIZ2.DOC	B:REPORT.DOC

Beachten Sie auch, daß, wenn Sie den Austauschparameter %0 in Ihrer Stapeldatei verwenden, dieser grundsätzlich durch die Laufwerksbezeichnung (falls angegeben) und den Dateinamen der Stapeldatei ersetzt wird. Wenn Sie nicht auf die Stapeldatei selbst Bezug nehmen wollen, beginnen Sie immer mit dem Austauschparameter %1.

5.7 Zusammenfassung

Sie haben in diesem Kapitel Einzelheiten zu folgenden internen und externen MS–DOS–Befehlen erfahren:

- FIND durchsucht eine Datei nach einer angegebenen Zeichenfolge.
- MORE erlaubt das Durchblättern auf dem Bildschirm in 23–Zeilen–Blöcken.
- PAUSE hält die Abarbeitung einer Stapeldatei an.
- REM fügte eine Kommentarzeile in eine Stapelverarbeitungsdatei ein.
- SORT sortiert eine Textdatei zeilenweise auf– oder absteigend.

Im nächsten Kapitel wird jeder MS–DOS–Befehl detailliert beschrieben.

6. Die MS–DOS–Befehle

6.1 Übersicht

Die in diesem Kapitel beschriebenen Befehle gliedern sich in zwei Teile:

1. Allgemeine MS–DOS–Befehle (Kapitel 6.3.1)

2. Stapelverarbeitungsbefehle (Kapitel 6.3.2)

In beiden Abschnitten sind die Befehle alphabetisch geordnet. Zum schnelleren Auffinden beginnt jeder Befehl auf einer neuen Seite.

Allgemeine Befehle können entweder intern oder extern sein. Externe Befehle sind in Dateien auf Diskette oder Festplatte gespeichert. Interne Befehle sind Bestandteil der Datei COMMAND.COM, die beim Start von MS–DOS in den Hauptspeicher des Computers geladen wird und dort verbleibt, bis der Computer abgeschaltet wird.

Einige der MS–DOS–Befehle arbeiten nicht in einem Computer– Netzwerk. Wenn Ihr Computer Station in einem Netzwerk ist und Sie einen solchen Befehl anwenden, erhalten Sie die Meldung

Cannot *Befehl* to a network device

(*Befehl* zu einer Netzwerkstation nicht erlaubt), wobei für *Befehl* der Name des jeweiligen Befehls gesetzt wird. Es sind dies die Befehle

CHKDSK	Überprüfung von Disk–Verzeichnissen;
DISKCOMP	Vergleich der Inhalte zweier Disketten;
DISKCOPY	Duplizierung von Disketten;
FDISK	Festplattenpartitionierung;
FORMAT	Formatierung von Disketten oder Festplatten;
LABEL	Erzeugen, Ändern oder Löschen von Diskkennungen;
RECOVER	Wiederherstellung von Dateien mit fehlerhaften Sektoren;
SUBST	Definition virtueller Laufwerke;
SYS	Übertragen der Systemdateien auf eine Disk.

6.2 Allgemeines Format der Befehlsbeschreibungen

Die Beschreibungen der einzelnen MS–DOS–Befehle sind in einem einheitlichen Format abgefaßt:

Format: Hier ist die exakte Schreibweise für einen Befehl angegeben. Dabei gelten die folgenden Syntax–Regeln:

1. Alle Wörter in Großbuchstaben sind MS–DOS-Schlüsselwörter, deren Schreibweise wie angegeben verbindlich ist. Bei der Eingabe über Tastatur können dabei Klein– oder Großbuchstaben verwendet werden, da MS–DOS diese Schlüsselwörter generell in Großbuchstaben umwandelt.

2. Alle Angaben in kursiver Groß– und Kleinschrift stellen Parameter dar, für die vom Anwender aktuelle Werte eingesetzt werden müssen.

3. Angaben zwischen eckigen Klammern ([ ]) sind wahlfrei.

4. Ein senkrechter Strich (|) trennt Alternativparameter. Es kann entweder der Parameter vor oder der hinter dem senkrechten Strich angegeben werden.

5. Eine Folge von Punkten (. . .) hinter einer Angabe bedeutet, daß diese Angabe, falls erforderlich, mehrmals spezifiziert werden kann.

6. Alle Sonderzeichen außer den eckigen Klammern und dem senkrechten Strich, also , ; – = ( ) ' , müssen da, wo sie stehen, auch angegeben werden.

7. Bei einigen Befehlsbeschreibungen paßt das allgemeine Format nicht auf eine Zeile, so daß es auf mehrere Zeilen aufgeteilt erscheint. In der Praxis muß jedoch ein MS–DOS–Befehl in einer Bildschirmzeile angegeben werden.

8. Bei der Befehlseingabe darf auch ein Pfadname verwendet werden. Befindet sich z.B. der Befehl ASSIGN in einem Verzeichnis namens BIN, so kann

```
\bin\assign c=e
```

eingegeben werden.

Typ: Hier wird der Typ des Befehls (intern oder extern) angegeben.

Bedeutung: Hier wird die Wirkung des jeweiligen Befehls beschrieben und die Parameter und ihre Definitionsbereiche werden erläutert.

Beispiele: Bei den meisten Befehlen wird die Befehlswirkung an einem oder mehreren Beispielen anschaulich erläutert.

Anmerkungen: Besondere Eigenschaften der einzelnen Befehle werden hier angegeben.

6.3 Die MS–DOS–Befehle im einzelnen

Vor der Beschreibung der einzelnen Befehle hier eine Übersicht über die im folgenden beschriebenen Befehle mit einem kurzen Stichwort über ihre Wirkung:

APPEND	setzt einen Suchpfad für Datendateien.
ASSIGN	weist einem Laufwerk einen anderen Namen zu.
ATTRIB	setzt oder zeigt Dateiattribute an.
BACKUP	sichert Festplattendateien auf Diskette.
BREAK	schaltet die Unterbrechungstasten–Überwachung ein oder aus.
CHDIR	wechselt das aktuelle Verzeichnis oder zeigt dessen Name an.
CHKDSK	analysiert das angegebene Verzeichnis und erzeugt einen Disk– und Speicherstatusbericht.
CLS	löscht den Bildschirm.
COMMAND	verarbeitet interne MS–DOS–Befehle.
COPY	kopiert Dateien.
CTTY	ändert oder setzt die Standard–Ein–/Ausgabekonsole.

DATE	zeigt und/oder ändert das aktuelle Datum.
DEL	löscht Dateien.
DIR	zeigt Disk–Verzeichniseinträge an.
DISKCOMP	vergleicht die Inhalte zweier Disketten.
DISKCOPY	dupliziert eine Diskette.
ERASE	wie DEL.
EXE2BIN	wandelt .EXE– in .COM oder .BIN–Dateien.
EXIT	verläßt den aktuellen Befehlsprozessor und geht zurück zur nächst höheren Befehlsebene.
FDISK	bereitet eine Festplatte für die Arbeit mit MS–DOS vor.
FIND	durchsucht eine Datei nach einer Zeichenkette.
FORMAT	formatiert eine Disk für die Datenspeicherung.
GRAFTABL	lädt eine Grafikzeichentabelle.
GRAPHICS	erlaubt graphische Ausgabe auf dem Drucker.
JOIN	ordnet ein Laufwerk einem Verzeichnis eines anderen Laufwerks zu.
KEYBxx	lädt den angegebenen nationalen Tastaturtreiber.
LABEL	erzeugt, ändert oder löscht eine Diskkennung.
MKDIR	erzeugt ein neues Unterverzeichnis.
MODE	setzt den Betriebsmodus für Paralleldrucker, Farbgrafikadapter und Adapter für Datenfernübertragung.
MORE	erlaubt das Durchblättern einer Textausgabe auf dem Bildschirm in 23–Zeilen–Blöcken.
PATH	setzt einen Befehls–Suchpfad.
PRINT	druckt Dateien im Hintergrund.
PROMPT	setzt eine neue Bereitschaftsanzeige.
RECOVER	stellt Diskettendateien mit fehlerhaften Sektoren wieder her.
RENAME	ändert den Namen einer Datei.
REPLACE	aktualisiert Dateien durch neuere Versionen.
RESTORE	speichert Dateien von Diskette auf Festplatte zurück, die mit BACKUP gesichert wurden.
RMDIR	löscht ein leeres Unterverzeichnis.
SET	fügt Zeichenfolgen in die Umfeld–Zeichentabelle des Befehlsprozessors ein.
SHARE	installiert die MS–DOS–Unterstützung für gemeinsamen Dateizugriff bei Netzwerk–Betrieb.
SORT	sortiert Textdateien zeilenweise alphabetisch auf– oder absteigend.
SUBST	definiert oder löscht ein virtuelles Laufwerk durch Zuweisung eines Pfadnamens zu einer Laufwerksbezeichnung.

SYS	überträgt die MS–DOS–Systemdateien vom aktuellen auf das angegebene Laufwerk.
TIME	zeigt und/oder ändert die aktuelle Uhrzeit.
TREE	zeigt den Pfad zu jedem Verzeichnis auf der angegebenen Disk und optional auch den Verzeichnisinhalt.
TYPE	zeigt den Inhalt einer Textdatei auf dem Bildschirm an.
VER	zeigt die aktuelle MS–DOS–Versionsnummer.
VERIFY	schaltet die Disk–Schreib–Überprüfung ein oder aus.
VOL	zeigt die Kennung der Disk im angegeben Laufwerk an.
XCOPY	kopiert Dateien und Unterverzeichnisse.

Stapelverarbeitungsbefehle

ECHO	schaltet die Anzeige der Befehle in der Stapeldatei ein oder aus oder zeigt Nachrichten an.
FOR. .IN. .DO	erlaubt Schleifenverarbeitung.
GOTO	erlaubt Verzweigungen in der Stapeldatei.
IF	erlaubt die bedingte Ausführung von Befehlen in der Stapeldatei.
PAUSE	hält die Abarbeitung einer Stapeldatei an.
REM	zeigt Kommentarzeilen aus einer Stapeldatei an.
SHIFT	läßt die Verwendung von mehr als 10 austauschbaren Parametern in einer Stapeldatei zu.

APPEND

Format: [*L*:][*BPfad*]**APPEND**[*Laufw*:][*Pfad*] [;[*Laufw*:] [*Pfad*]. . .]

Typ: extern

Bedeutung: Setzt einen Suchpfad für Datendateien.

L*:*BPfad Laufwerk und Suchpfad für die APPEND–Befehlsdatei.

***Laufw*:** Eine gültige Laufwerksbezeichnung nach MS–DOS–Konventionen für die Disk, für die der Suchpfad gesetzt werden soll. Wird ***Laufw*:** weggelassen, bezieht sich der Suchpfad auf das aktuelle Laufwerk.

Pfad Der zu setzende Suchpfad (s.a. Kapitel 4.2.1), der das nach einer Datendatei zu durchsuchende Verzeichnis angibt. Wird ***Pfad*** weggelassen, so wird der aktuelle Suchpfad für das aktuelle oder das angegebene Laufwerk angezeigt.

Wird der Befehl in der Form **APPEND ;** verwendet, so setzt MS–DOS den NUL–Pfad. D.h., daß dann nur das aktuelle Verzeichnis nach Datendateien durchsucht wird.

Beispiel: Angenommen, sie wollen auf Datendateien in einem Unterverzeichnis namens BRIEFE auf Laufwerk B und in einem Unterverzeichnis namens REPORTS auf Laufwerk A zugreifen. Der entsprechende APPEND–Befehl lautet

```
append b:briefe;a:reports
```

Anmerkungen:

- Sie können mehrere Suchpfade mit einem APPEND–Befehl angeben, wenn sie diese durch Semikolon (;) trennen.

- APPEND sucht die Datenpfade für alle Dateien, unabhängig von deren Namenserweiterung nur mit den folgenden MS–DOS–Systemaufrufen:

Code	Funktion
0FH	Datei öffnen (FCB)
23H	Dateigröße lesen
3DH	beliebige Datei öffnen

- Mit APPEND können in einem Netzwerk auch Dateien bei anderen Teilnehmern gesucht werden.

- Wenn der MS–DOS–Befehl APPEND zusammen mit dem Befehl ASSIGN (s. dort) verwendet wird, so muß **zuerst** der APPEND–Befehl eingegeben werden.

ASSIGN

Format: [*L*:][*BPfad*]**ASSIGN** [*x*[=]*y*[. . .]]

Typ: extern

Bedeutung: Weist MS–DOS an, für alle nachfolgenden Disk– oder Dateireferenzen einen Laufwerksbuchstaben durch einen anderen zu ersetzen. Mit anderen Worten, alle nachfolgenden Diskzugriffe sollen nicht auf das im jeweiligen Befehl angegebene Laufwerk, sondern auf das durch diesen Befehl dafür ersetzte erfolgen.

L*:*BPfad Laufwerk und Suchpfad für die ASSIGN–Befehlsdatei.

x Das Laufwerk, auf das bisher alle Zugriffe erfolgten

y Das Laufwerk, auf das zukünftig alle Zugriffe erfolgen sollen.

Werden die Parameter weggelassen, so werden alle Zuweisungen wieder aufgehoben.

Beispiele:

```
assign a=c
```

Alle Zugriffe auf Laufwerk A gehen künftig auf Laufwerk C (Festplatte).

```
assign a=c b=c
```

Alle Zugriffe auf die Laufwerke A und B gehen künftig auf Laufwerk C (Festplatte).

Anmerkungen:

- Das Gleichheitszeichen (=) braucht nicht angegeben zu werden.
- Nach den Laufwerksbezeichnungen darf **kein** Doppelpunkt (:) angegeben werden.

- Der ASSIGN–Befehl sollte nur in Verbindung mit Anwenderprogrammen und darf nicht zusammen mit dem BACKUP– oder PRINT–Befehl benutzt werden. Ebenso sollte er nicht bei der normalen Arbeit mit MS–DOS verwendet werden, da damit der tatsächliche Laufwerkstyp den Befehlen verborgen bleibt, die aktuelle Laufwerksinformationen benötigen. Die Befehle FORMAT und DISKCOPY ignorieren jegliche Laufwerkszuweisungen mit ASSIGN.

- Es können nur physikalisch vorhandene Laufwerke zugewiesen werden.

ATTRIB

Format: [*L*:][*BPfad*]**ATTRIB** [+**R**|–**R**] [+**A**|–**A**] [*Laufw*:][*Pfad*]*Datei*[*.Erw*]

Typ: extern

Bedeutung: Setzt oder löscht das Schreibschutz- oder das Archiv-Attribut für eine Datei oder zeigt deren Attribute an.

L*:*BPfad Laufwerk und Suchpfad für die ATTRIB–Befehlsdatei.

+R|–R Setzt (+R) oder löscht (–R) das Schreibschutzattribut (read only = nur lesen).

+A|–A Setzt (+A) oder löscht (–A) das Archivattribut.

***Laufw*:** Eine gültige Laufwerksbezeichnung nach MS–DOS–Konventionen für die Disk, auf der die bezogene Datei gespeichert ist. Wird ***Laufw*:** weggelassen, so nimmt MS–DOS das aktuelle Laufwerk an.

Pfad Der Pfad (s.a. Kapitel 4.2.1), auf dem die bezogene Datei zu finden ist. Wird ***Pfad*** weggelassen, so wird die bezogene Datei im aktuellen Verzeichnis gesucht.

***Datei*[*.Erw*]** Eine gültige Dateibezeichnung nach MS–DOS–Konventionen (s.a. Kapitel 4.1.5), ggf. mit Namenserweiterung.

Wird als Parameter nur der Dateiname, ggf. zusammen mit seinem Suchpfad und Laufwerk, angegeben, so werden die für diese Datei gültigen Attribute angezeigt.

Beispiele:

```
attrib +r brief.txt
```

Die Datei BRIEF.TXT im aktuellen Verzeichnis der Disk im aktuellen Laufwerk wird schreibgeschützt (R wie read only = nur Lesen).

```
attrib -r c:\prog\sortier.bas
```

Der Schreibschutz der Datei SORTIER.BAS im Unterverzeichnis PROG auf der Festplatte in Laufwerk C wird aufgehoben.

```
attrib *.com
```

Die Attribute aller Dateien mit der Namenserweiterung .COM werden angezeigt; z.B.:

```
A  \WORD.COM
A  \COMMAND.COM
A  \ATTRIB.COM
```

Der Buchstabe A bedeutet, daß die Datei WORD.COM eine Archivdatei ist.

Anmerkungen:

- Wenn ein Anwenderprogramm eine Datei mit Lese– und Schreiberlaubnis eröffnet, sollte mit dem ATTRIB–Befehl die Schreiberlaubnis aufgehoben werden, um Dateizugriffe in einem Netzwerk zu erlauben.

- Die Befehle BACKUP und RESTORE verwenden das Archiv–Attribut für die Steuerung einer selektiven Sicherung und Rückspeicherung bei Dateien, die zwischendurch verändert wurden. Mit Hilfe der Parameter +A und –A können Dateien ausgewählt werden, die mit dem BACKUP–Befehl zusammen mit dem /M–Schalter (s. Backup–Befehl) gesichert werden sollen.

- Wenn eine Datei mit dem Schreibschutz–Attribut mit COPY kopiert wird, ist die Kopie **nicht** mehr schreibgeschützt. Der DISKCOPY–Befehl bewahrt dagegen ggf. gesetzte Attribute.

BACKUP

Format: [*L*:][*BPfad*]**BACKUP** [*QLaufw*:][*Pfad*] [*Datei*[.*Erw*]] *ZLaufw*: [/**S**] [/**M**] [/**A**] [/**D**:*mm–tt–jj*] [/**T**:*hh.mm*] [/**L**:[*Logdatei*]]

Typ: extern

Bedeutung: Erzeugt Sicherungsdateien vom Inhalt der angegebenen Festplatte oder Diskette.

L*:*BPfad Laufwerk und Suchpfad für die BACKUP–Befehlsdatei.

***QLaufw*:** Eine gültige Laufwerksbezeichnung nach MS–DOS–Konventionen für die Disk, von der Dateien gesichert werden sollen. Wird ***QLaufw*:** weggelassen, so nimmt MS–DOS das aktuelle Laufwerk an.

Pfad Der Pfad (s.a. Kapitel 4.2.1), auf dem die zu sichernde Datei zu finden ist. Wird ***Pfad*** weggelassen, so wird die zu sichernde Datei im aktuellen Verzeichnis gesucht.

***Datei*[.*Erw*]** Eine gültige Dateibezeichnung nach MS–DOS–Konventionen (s.a. Kapitel 4.1.5), ggf. mit Namenserweiterung. Wird dieser Parameter weggelassen, so werden alle Dateien des aktuellen oder angegebenen Verzeichnisses gesichert.

***ZLaufw*:** Eine gültige Laufwerksbezeichnung nach MS–DOS–Konventionen für die Disk, auf die die Dateien gesichert werden sollen.

/S Wird dieser Schalter angegeben, so werden die Inhalte aller Unterverzeichnisse zum angegebenen oder aktuellen Verzeichnis ebenfalls gesichert.

/M Wird dieser Schalter angegeben, so werden nur solche Dateien gesichert, die seit dem letzten BACKUP verändert wurden.

/A Wird dieser Schalter angegeben, so werden die gesicherten Dateien den ggf. bereits auf der BACKUP-Disk befindlichen hinzugefügt.

/D:*mm–tt–jj* Wird der Datumsschalter zusammen mit einem im angegebenen Format spezifizierten Datum gesetzt, so werden nur solche Dateien gesichert, die am angegebenen Datum oder später verändert wurden. Das Datumsformat wird durch den COUNTRY–Befehl in der CONFIG.SYS–Datei festgelegt (s. Kapitel 2.3).

/T:*hh.mm* Wird der Zeitschalter zusammen mit einer im angegebenen Format spezifizierten Zeit gesetzt, so werden nur solche Dateien gesichert, die zur angegebenen Zeit oder später verändert wurden.

/L:*Logdatei* Wird dieser Schalter zusammen mit einem Dateinamen angegeben, so erzeugt der BACKUP–Befehl in der angegebenen Log–Datei ein Protokoll des BACKUP–Vorganges. Wird ***Logdatei*** weggelassen, so erzeugt BACKUP eine Datei namens BACKUP.LOG im Basisverzeichnis der Disk im Quellaufwerk. Diese Datei enthält als ersten Eintrag Datum und Uhrzeit des Beginns des BACKUP–Vorganges und dann alle Namen der gesicherten Dateien. Jedem Namen ist außerdem die Nummer der BACKUP–Disk vorangestellt, auf der sich die angegebene Datei befindet. Diese Information kann zum Rückschreiben einer bestimmten Datei mit dem RESTORE–Befehl (s. dort) benutzt werden, damit RESTORE nicht lange suchen muß. Existiert die angegebene Log–Datei bereits, so werden die neuen Informationen an die bereits vorhandenen angefügt.

Beispiel:

Die Benutzerin Maria möchte alle Dateien im Verzeichnis \BENUTZER\MARIA auf der Festplatte im Laufwerk C auf eine leere, formatierte Diskette im Laufwerk A sichern. Dann muß sie folgenden Befehl eingeben:

```
backup c:\benutzer\maria a:
```

Anmerkungen:

- Der BACKUP–Befehl sichert Dateien von:
  - Festplatte auf Diskette
  - Diskette auf Diskette
  - Diskette auf Festplatte
  - Festplatte auf Festplatte
- Mit BACKUP können auch Dateien von einer Diskette auf eine andere Diskette mit unterschiedlicher Sektor/Spur–, Spur– oder Seiten–Anzahl gesichert werden.
- Wenn nicht der /A–Schalter verwendet wird, überschreibt BACKUP ggf. auf der Zieldisk vorhandene Dateien.
- Während des Sicherungslaufes zeigt BACKUP die Namen der gesicherten Dateien auf dem Bildschirm an.
- Reicht der Platz auf der Zieldisk für die Sicherung nicht aus, so fordert BACKUP weitere Disketten an. Sie sollten diese mit aufeinanderfolgenden Nummern versehen, um das Rückschreiben mit RESTORE (s. dort) zu erleichtern.
- Wenn gemeinsamer Dateizugriff in einem Netzwerk installiert ist, können mit BACKUP nur die Dateien gesichert werden, auf die Sie als Benutzer Zugriff haben.
- Haben Sie mit den Befehlen ASSIGN, JOIN oder SUBST (s. dort) die Laufwerksbezeichnungen anders zugewiesen, sollten Sie den BACKUP–Befehl nicht verwenden, da Sie dann die gesicherten Dateien möglicherweise nicht mehr mit RESTORE wiederherstellen können.
- Der BACKUP–Befehl entspricht **nicht** dem COPY-Befehl. Ersterer erzeugt in der Sicherungsdatei Steuercodefolgen, die der RESTORE–Befehl für die Wiederherstellung der Datei benötigt, während letzterer ein genaues Abbild der Quelldatei erzeugt.

- Der BACKUP–Befehl erzeugt bei der Beendigung die folgenden Fehlergrad–Codes, die mit dem Stapelverarbeitungsbefehl IF ERRORLEVEL (s. dort in Kapitel 6.3.2) verarbeitet werden können:

  0 normale Beendigung; kein Fehler.
  1 keine zu sichernden Dateien gefunden.
  2 auf Grund von Konflikten beim gemeinsamen Dateizugriff in einem Netz konnten einige Dateien nicht gesichert werden.
  3 Abbruch durch den Benutzer mit Ctrl–Break.
  4 Abbruch durch Fehler.

- Wird als Zieldisk eine Festplatte angegeben, so werden die Sicherungsdateien im Unterverzeichnis \BACKUP abgelegt; bei einer Diskette dagegen im Basisverzeichnis.

BREAK

Format: **BREAK [ON|OFF]**

Typ: intern

Bedeutung: Schaltet die Überwachung der Tastenkombination Ctrl–Break oder Ctrl–C ein (BREAK ON) oder aus (BREAK OFF). Wird BREAK ohne Parameter verwendet, so wird der gegenwärtige Zustand der Überwachung angezeigt.

Anmerkungen:

- Der Standardwert, den MS–DOS beim Start setzt, ist OFF. In diesem Fall überwacht MS–DOS die Ctrl–Break–Taste nur bei Leseoperationen von der Tastatur und bei Schreiboperationen zum Bildschirm oder Drucker. Um die Überwachung auch auf andere Operationen wie z.B. Schreib–/Lesezugriffe auf Disk zu erweitern, muß BREAK ON gesetzt werden.

- Einige Anwenderprogramme können selbst die Überwachung der Ctrl–Break–Taste beinhalten. Diese Programme werden durch den BREAK ON|OFF–Befehl **nicht** beeinflußt.

CHDIR

Format: **CHDIR** [[***Laufw***:]***Pfad***]

oder

CD [[***Laufw***:]***Pfad***]

Typ: intern

Bedeutung: Wechselt das aktuelle Verzeichnis oder zeigt es an.

Laufw: Eine gültige Laufwerksbezeichnung nach MS–DOS–Konventionen für die Disk, auf der das Verzeichnis gändert oder angezeigt werden soll. Wird ***Laufw***: weggelassen, so nimmt MS–DOS das aktuelle Laufwerk an.

Pfad Der Pfad (s.a. Kapitel 4.2.1), auf dem das bezogene Verzeichnis zu finden ist. Wird ***Pfad*** weggelassen, so zeigt MS–DOS vom aktuellen oder anegebenen Laufwerk das aktuelle Verzeichnis zusammen mit seinem Suchpfad an.

Wird der Befehl in der Form CHDIR . . oder CD . . angewendet, so wird das Vaterverzeichnis zum aktuellen Verzeichnis angezeigt.

Beispiele: Ist z.B. das Verzeichnis \BENUTZER\MARIA auf Laufwerk A: das aktuelle Verzeichnis, so ergibt der Befehl

```
chdir a:
```

die Anzeige

```
a:\benutzer\maria
```

Der Befehl

```
cd \rudolf\progr
```

macht das Verzeichnis \RUDOLF\PROGR auf der Disk im aktuellen Laufwerk zum aktuellen Verzeichnis.

Der Befehl

```
cd \
```

aktualisiert das Basis–Verzeichnis auf der Disk im aktuellen Laufwerk.

CHKDSK

Format: [*L*:][*BPfad*]**CHKDSK** [*Laufw*:][*Pfad*] [*Datei*[*.Erw*]] [**/F**] [**/V**]

Typ: extern; bei Netzwerkbetrieb nicht erlaubt.

Bedeutung: Analysiert das Inhaltsverzeichnis und die Dateizuordnungstabelle der angegebenen Disk und erzeugt einen Statusreport (ggf. auch für die angegebene Datei).

L*:*BPfad Laufwerk und Suchpfad für die CHKDSK–Befehlsdatei.

***Laufw*:** Eine gültige Laufwerksbezeichnung nach MS–DOS–Konventionen für die Disk, für die der Statusreport erzeugt werden soll. Wird ***Laufw*:** weggelassen, so nimmt MS–DOS das aktuelle Laufwerk an.

Pfad Datei.Erw Der Pfad (s.a. Kapitel 4.2.1), auf dem die angegebene Datei zu finden ist, für die der Statusreport erstellt werden soll. Wird keine Datei angegeben, so wird der Statusbericht für die gesamte Disk erstellt.

/F Wird dieser Schalter angegeben, so korrigiert CHDISK Fehler im Verzeichnis oder in der Dateizuordnungstabelle automatisch. Wird er **nicht** angegeben, so zeigt CHKDSK die Fehler an, korrigiert sie aber nicht.

/V Wird dieser Schalter angegeben, so werden während der Überprüfung alle Verzeichnisse und deren Inhalte mit ihren Pfaden angezeigt.

Ein typischer Statusbericht von der Festplatte könnte so aussehen:

```
21309440 bytes total disk space
   45056 bytes in 3 hidden files
  661504 bytes in 53 user files
20602880 bytes available on disk

  524288 bytes total memory
  376128 bytes free
```

Beispiel:

Wenn Sie den Statusreport des CHKDSK–Befehls von Laufwerk A für zukünftige Verwendung speichern wollen, können Sie die Ausgabe von CHKDSK in eine Datei umleiten, indem Sie folgendes eingeben:

```
chkdsk a:>dskreprt
```

Alle Fehleranzeigen werden dann in diese Datei auf dem aktuellen Laufwerk geschrieben. Denken Sie aber daran, in diesem Fall **nicht** den /F–Schalter zu benutzen.

Wenn Sie Fehler in den Verzeichnissen oder der Dateizuordnungstabelle von der Disk in Laufwerk a: korrigieren wollen, geben Sie ein:

```
chkdsk a:/f
```

Anmerkungen:

- Um die Integrität Ihrer Disk–Medien zu garantieren, sollten Sie gelegentlich von jeder Festplatte und wichtigen Diskette mit CHKDSK einen Statusreport erstellen.

- Bei Systemanfragen von CHKDSK, die mit Y oder N (Yes oder No) zu beantworten sind, muß aus Sicherheitsgründen zusätzlich die Eingabetaste gedrückt werden.

- Wenn ein Statusreport für eine bestimmte Datei erstellt werden soll, so enthält dieser auch eine Liste aller nicht zusammenhängender Diskbereiche. Je stärker eine Datei fragmentiert ist, umso mehr Zeit braucht MS–DOS für den Zugriff, da nicht sequentiell gelesen werden kann.

- Kann CHKDSK Diskbereiche keiner Datei zuordnen, so fragt es an, ob diese verlorenen Bereiche in Dateien wiederhergestellt werden sollen. Wird mit Y für Yes (Ja) geantwortet und war der /F–Schalter angegeben, so schreibt CHKDSK jede Folge solcher Bereiche in eine Datei im Basisverzeichnis des angegebenen Laufwerks, die den Namen FILEnnnn.CHK erhält, wobei nnnn die Folgenummer ist, die mit 0000 beginnt.

- CHKDSK wartet bei Disketten nicht, bis diese eingelegt ist.

- Wenn nicht der /F–Schalter angegeben wurde, korrigiert CHKDSK keinerlei Fehler, die es bei der Analyse findet, sondern zeigt diese nur an. Im Anhang A sind alle MS–DOS–Fehlermeldungen zusammengestellt und beschrieben.

CLS

Format: **CLS**

Typ: intern

Bedeutung: löscht den Bildschirm, indem die ANSI–ESCAPE–Sequenz ESC[2J an den Bildschirm gesendet wird. Wurde bei einem Farbbildschirm Vorder– und Hintergrundfarbe gewählt, so bleiben beide Farben unverändert.

COMMAND

Format: [*L*:][*BPfad*]**COMMAND** [*Laufw*:][*Pfad*] [*Gerät*] [**/E:***nnnnn*] [**/P**] [**/C** *Befehl*]

Typ: extern

Bedeutung: Startet einen neuen MS-DOS-Befehlsprozessor. Der Befehlsprozessor enthält alle **internen** MS-DOS-Befehle.

L:BPfad Laufwerk und Suchpfad für die COMMAND-.COM–Befehlsdatei.

Laufw: Eine gültige Laufwerksbezeichnung nach MS–DOS–Konventionen für die Disk, auf der der zu ladende Befehlsprozessor gespeichert ist. Wird ***Laufw:*** weggelassen, so nimmt MS–DOS das aktuelle Laufwerk an.

Pfad Der Pfad (s.a. Kapitel 4.2.1), auf dem der zu ladende Befehlsprozessor zu finden ist. Wird ***Pfad*** weggelassen so sucht MS–DOS im aktuellen Verzeichnis.

Gerät Ein anderes Gerät für Ein– und Ausgabe (z.B. AUX) als die Standard–Ein–/Ausgabe.

E:*nnnnn* Die Größe der Umfeld–Zeichentabelle in Bytes im Bereich zwischen 128 und 32768. Voreingestellt ist 128. Wird ein kleinerer Wert als 128 angegeben, so zeigt MS–DOS die Fehlermeldung

```
Invalid environment size specified
```

(ungültige Umgebungsgröße angegeben). Wird ein größerer Wert als 32768 angegeben, so wird dieselbe Fehlermeldung angezeigt, die Tabelle aber mit 32768 Bytes dimensioniert.

/P Wird dieser Schalter angegeben, so kann COMMAND.COM nicht zu einem übergeordneten Befehlsprozessor verlassen werden.

/C Befehl Wenn der /C–Schalter benutzt wird, muß er als letzter Parameter in der Argumentliste angegeben werden. Der oder die für ***Befehl*** angegebene(n) Befehl(e) wird/werden gleich ausgeführt.

Beispiel: Mit der folgenden Befehlsfolge weisen Sie MS–DOS an, die folgenden Arbeiten auszuführen:

- Start eines neuen Befehlsprozessors unterhalb des gerade aktiven.
- Start des CHKDSK–Programmes für Laufwerk A.
- Rückkehr zum übergeordneten Befehlsprozessors.

```
command /c chkdsk a:
exit
```

Anmerkungen:

- Im System–Konfigurierungsbeispiel im Kapitel 2.4 ist die Verwendung eines Pfadnamens und des /P–Schalters bei der Spezifizierung eines Befehlsprozessors beschrieben.

- Mit dem Start eines neuen Befehlsprozessors wird auch eine neue Umfeldzeichentabelle eingerichtet, die eine Kopie des Vater–Befehlsprozessors ist. Die Größe dieser Tabelle kann jedoch mit dem /E–Schalter geändert werden, ohne die alte zu beeinflussen.

- Der Befehlsprozessor wird in zwei Teilen in den Hauptspeicher geladen, den **transienten** und den **residenten** Teil. Es gibt Anwenderprogramme, die den transienten Teil im Speicher überschreiben. In diesem Fall sucht der residente Teil nach Beendigung des Programmes den transienten Teil auf Disk und lädt ihn neu.

COPY

Format:

COPY [/A][/B][*Laufw*:][*Pfad*]
Datei[.*Erw*][/A][/B] [*Laufw*:][*Pfad*]
[*Datei*[.*Erw*]][/A][/B][/V]

oder

COPY [/A][/B][*Laufw*:][*Pfad*]
Datei[.*Erw*][/A][/B][+[*Laufw*:][*Pfad*]
[*Datei*[.*Erw*]][/A][/B]. . .]
[*Laufw*:][*Pfad*][*Datei*[.*Erw*]][/A][/B][/V]

oder

COPY [/A][/B][*Laufw*:][*Pfad*]
Datei[.*Erw*][/A][/B][+[[,,]*Laufw*:][*Pfad*]
[*Datei*[.*Erw*]][/A][/B]. . .]
[*Laufw*:][*Pfad*][*Datei*[.*Erw*]][/A][/B][/V]

Typ:

intern

Bedeutung:

kopiert die angegebene(n) Datei(en) auf die angegebene Disk oder fügt mehrere Dateien zu einer neuen zusammen.

Laufw*:*PfadDatei.Erw Die vollständigen Angabe für die Quelldatei (d.i. die erste Dateiangabe(n) für die zu kopierende(n) Datei(en)) und die Zieldatei (d.i. die zweite Dateiangabe für die Datei, in die kopiert werden soll). Wird die Angabe für die Zieldatei weggelassen, so erhält die Kopie den Namen der Quelldatei. Werden mehrere Quelldateiangaben durch das Pluszeichen (+) miteinander verbunden, so werden diese Dateien in der angegebenen Reihenfolge in die Zieldatei kopiert (Dateiverkettung).

/A und **/B** Diese beiden Schalter kennzeichnen die Daten– und Kopier–Art und beziehen sich auf die vorhergehende und alle nachfolgenden Dateiangaben, bis ein neuer Schalter /A oder /B gesetzt wird. Je nachdem, ob

diese Schalter bei der/den Quelldatei(en) oder der Zieldatei angegeben werden, haben sie unterschiedliche Bedeutung.

Bei der/den Quell–Datei(en):

/A Die Datei wird als ASCII (Text)–Datei behandelt und alle Daten werden bis zum Auftreten der ersten Datei-Endmarke (bei EDLIN–Dateien ist dies das Ctrl–Z–Zeichen (1AH)), jedoch ausschließlich dieser Marke, kopiert.

/B Die Datei wird als Binärdatei behandelt und in ihrem gesamten im Diskverzeichnis vermerkten Umfang kopiert.

Bei der Zieldatei:

/A Als letztes Zeichen wird das Ctrl–Z–Zeichen (1AH) als Datei-Endmarke in die Zieldatei geschrieben (Voreingestellt beim Kopieren mit Verkettung).

/B In die Zieldatei wird keine Datei-Endmarke (1AH) geschrieben (Voreingestellt beim Kopieren ohne Verkettung).

/V Wird dieser Schalter angegeben, so werden die kopierten Daten mit den Quelldaten während des Kopiervorganges verglichen, um die Integrität der kopierten Daten zu gewährleisten. Dieser Schalter hat dieselbe Bedeutung wie der Befehl VERIFY ON (s. dort in diesem Kapitel). Wird er angegeben, so verlangsamt sich die Ausführung des COPY–Befehls.

Beispiele:

Wegen der Mächtigkeit des COPY–Befehls werden die Beispiele zur Verdeutlichung je nach Benutzung des COPY–Befehls gegliedert:

1. Kopieren unter Beibehaltung des Dateinamens

In diesem Fall müssen Quell– und Ziel–Laufwerk unterschiedlich sein, es sei denn, die Zieldatei wird in ein anderes Verzeichnis kopiert. Datums– und Uhrzeit–Eintrag der Quelldatei werden in die Zieldatei übernommen.

Wenn das aktuelle Laufwerk C ist, wird mit dem folgenden Befehl die Datei REPORT.TXT von der Disk in Laufwerk A unter Beibehaltung des Namens in das aktuelle Verzeichnis der Festplatte kopiert:

```
copy a:report.txt
```

Mit dem folgenden Befehl werden alle Dateien aus dem aktuellen Verzeichnis der Disk im aktuellen Laufwerk in das aktuelle Verzeichnis der Disk in Laufwerk A kopiert:

```
copy *.* a:
```

Dabei werden die Dateinamen während des Kopiervorganges angezeigt. Mit diesem Befehl lassen sich auf einfache Weise auch stark fragmentierte Dateien einer Disk wieder effektiv speichern.

Der folgende Befehl kopiert die Datei REPORT.TXT aus dem Basisverzeichnis der Disk in Laufwerk A: in das Unterverzeichnis MARIA unter Beibehaltung des Namens auf demselben Laufwerk:

```
copy a:\report.txt \maria
```

Bei diesem Befehl geht MS–DOS davon aus, daß das Unterverzeichnis MARIA auch existiert. Falls nicht, würde die Datei REPORT.TXT in eine andere Datei namens MARIA kopiert.

2. Kopieren mit Vergabe eines anderen Namens

In diesem Fall brauchen Quell– und Ziel–Laufwerk nicht verschieden zu sein. Eine Datei kann also auf demselben Laufwerk und im selben Verzeichnis umkopiert werden. Datums– und Uhrzeit–Eintrag der Quelldatei werden in die Zieldatei übernommen.

Der folgende Befehl kopiert die Datei REPORT.TXT aus dem aktuellen Verzeichnis der Disk im aktuellen Lauf-

werk in das aktuelle Verzeichnis der Disk in Laufwerk A und gibt der Kopie den Namen REPORT.RPT:

```
copy report.txt a:*.rpt
```

Als Dateinamen dürfen auch die für MS–DOS reservierten Gerätenamen (s.a. Kapitel 4.1.6) verwendet werden. Z.B. zeigt der folgende Befehl den Inhalt der AUTOEXEC.BAT–Datei auf dem Bildschirm an, vorausgesetzt, diese Datei existiert im aktuellen Verzeichnis der Disk im aktuellen Laufwerk:

```
copy autoexec.bat con:
```

Genausogut kann die Datei REPORT.TXT aus dem Verzeichnis MARIA auf der Disk im Laufwerk A über die Datenfernübertragungsschnittstelle mit dem folgenden Befehl übertragen werden:

```
copy a:\maria\report.txt aux:
```

Um den Empfang von der Datenfernübertragungsschnittstelle an den Drucker zu senden, brauchen Sie nur einzugeben:

```
copy aux: lpt1:
```

Eine einfache Textdatei wie die Datei AUTOEXEC.BAT können Sie mit dem COPY–Befehl auf folgende Art und Weise erstellen:

```
copy con: autoexec.bat
```

Geben Sie dann den gewünschten Text zeilenweise ein und drücken Sie nach jeder Zeile die Eingabe–Taste. Abschließend drücken Sie die Funktionstaste F6 oder die Tastenkombination Ctrl–Z, womit eine Datei-Endmarke in die Zieldatei geschrieben wird (s.a. Kapitel 7.2) und drücken Sie dann noch einmal die Eingabetaste. Damit wird die Datei ordnungsgemäß abgeschlossen.

3. Verketten mehrerer Quelldateien in einer Zieldatei

Bei der Dateiverkettung wird zunächst die erste Quelldatei in die Zieldatei kopiert, anschließend die nächste Quelldatei an das Ende der Zieldatei angefügt, usw. In diesem Fall wird bei der Zieldatei das aktuelle Datum und die aktuelle Uhrzeit eingetragen, da sie ja gegenüber den Quelldateien verändert wurde. Die Dateiangaben für die beliebig vielen Quelldateien werden bei dieser Variante des COPY–Befehls mit dem Plus–Zeichen (+) voneinander getrennt.

Der folgende Befehl verkettet die Dateien REPORT1.TXT und REPORT2.TXT aus dem aktuellen Verzeichnis der Disk im Laufwerk A sowie die Datei BRIEF.TXT aus dem Unterverzeichnis KORRESP auf der Disk in Laufwerk B zur Datei PROJEKT.TXT im aktuellen Verzeichnis der Disk im aktuellen Laufwerk:

```
copy a:report1.txt+a:report2.txt+b:
\korresp\brief.txt projekt.txt
```

Soll eine Datei an das Ende einer anderen Datei angefügt werden, so wird die Zieldateibezeichnung einfach weggelassen. Im folgenden Beispiel wird die Datei BRIEF.TXT an das Ende der Datei REPORT.TXT angehängt und das Ergebnis in der Datei REPORT.TXT belassen:

```
copy report.txt+brief.txt
```

Um alle Dateien mit den Namenserweiterungen .TXT und .RPT im aktuellen Verzeichnis der Disk im aktuellen Laufwerk in einer Datei SAMMEL.TXT ebenda zusammenzufügen, genügt der Befehl:

```
copy *.txt+*.rpt sammel.txt
```

Soll eine Datei nicht verändert werden, sondern nur das aktuelle Datum und die Uhrzeit fortgeschrieben werden,

so kann z.B. für die Datei AUTOEXEC.BAT im Laufwerk A folgender Befehl eingegeben werden:

```
copy autoexec.bat+,, a:
```

Anmerkungen:

- Der COPY–Befehl entspricht **nicht** dem BACKUP-Befehl. BACKUP erstellt Sicherungskopien von allen Dateien einschließlich der Unterverzeichnisse. COPY kopiert nur die angegebenen Dateien aus dem aktuellen oder dem angegebenen Verzeichnis (s.a. XCOPY–Befehl in diesem Kapitel).

- Die Verwendung der Jokerzeichen * und ? zur Definierung globaler Dateinamen sind erlaubt.

- Verwenden Sie **nicht** die Verkettungsvariante des COPY–Befehls, wenn eine der Quelldateien dieselbe Namenserweiterung hat wie die Zieldatei. Z.B. ist der folgende Befehl falsch:

  ```
  copy *.txt alles.txt
  ```

  Zunächst würde MS–DOS hier den Fehler nicht diagnostizieren, solange an die Datei ALLES.TXT keine andere Datei angehängt wird. Dann jedoch könnte die Datei ALLES.TXT bereits verändert worden sein. COPY vergleicht in einem solchen Fall zunächst den Namen der Quelldatei mit dem der Zieldatei. Sind sie gleich, wird diese Quelldatei übergangen und die Meldung

  ```
  Content of destination lost before copy
  ```

  (Inhalt der Zieldatei vor dem Kopieren verloren) angezeigt. Der weitere Kopiervorgang wird dann normal fortgesetzt. So werden mit dem folgenden Befehl alle Dateien namens *.txt (mit Ausnahme der Datei ALLES.TXT selbst) zur Datei ALLES.TXT verkettet, ohne daß eine Meldung angezeigt wird:

  ```
  copy alles.txt+*.txt
  ```

- Wenn zu oder von einem Gerät kopiert wird, für das MS–DOS einen Gerätenamen reserviert hat (s.a. Kapitel 4.1.6), so nimmt MS–DOS einen ASCII-Modus an (/A–Schalter), es sei denn, es wurde der /B–Schalter angegeben.

- Ist die Quelldatei schreibgeschützt (Attribut R (read only)), so wird dieses Attribut **nicht** mit kopiert.

- Werden Dateien mit COPY verkettet, so muß mindestens eine der angegebenen Quelldateien, nicht jedoch zwingend alle, vorhanden sein.

CTTY

Format: **CTTY *Gerät***

Typ: intern

Bedeutung: Ersetzt das Standard–Ein– und –Ausgabegerät (Tastatur und Bildschirm) durch das angegebene Gerät oder setzt Tastatur und Bildschirm wieder als Standard–Ein–/Ausgabegerät.

Gerät Der Name für das Gerät, das als Primärkonsole dienen soll. Wird für ***Gerät*** der Name CON gesetzt, so benutzt MS–DOS wieder Tastatur und Bildschirm als Standard–Ein–/Ausgabe–Gerät.

Beispiel: Der folgende Befehl benutzt die Datenfernübertragungsschnittstelle als Standard–Ein–/Ausgabe–Gerät:

```
ctty aux
```

Anmerkungen:

- Der CTTY–Befehl erlaubt auch die Angabe anwenderspezifischer Gerätenamen, damit der Benutzer auch eigene Gerätetreiber verwenden kann, die jedoch installiert sein müssen (s.a. Kapitel 2). Es muß jedoch sichergestellt sein, daß das angegebene Gerät sowohl Ein– als auch Ausgaben verarbeiten kann. LPT1: z.B. kann nicht angegeben werden, da von dieser Einheit nicht gelesen werden kann.

- Viele Anwenderprogramme benutzen für Ein– und/oder Ausgabe nicht die Standard–MS–DOS–Systemaufrufe sondern empfangen oder senden ihre Ein– oder Ausgaben direkt über die Hardware-Schnittstellen im Computer. Auf solche Programme haben Zuweisungen mit dem CTTY–Befehl keinen Einfluß.

DATE

Format: **DATE** [*tt.mm.*[**19**|**20**]*jj*]

Typ: intern

Bedeutung: Setzt das System–Datum auf das angegebene Datum oder zeigt das aktuelle Systemdatum an.

tt Eine Zahl zwischen 1 und 31 (eine oder zwei Ziffern) für den aktuellen Tag.

mm Eine Zahl zwischen 1 und 12 (eine oder zwei Ziffern) für den aktuellen Monat.

jj Eine Zahl zwischen 80 und 99 für die Jahre 1980 bis 2099. Wird die Angabe 19 oder 20 vor der Jahreszahl weggelassen, so nimmt MS–DOS 19 an.

Anmerkungen:

- Wird der DATE–Befehl ohne Parameter angegeben, so zeigt MS–DOS die Meldung

  ```
  Current date is tag tt-mm-19jj
  Enter new date (dd-mm-yy)_:
  ```

  (Aktuelles Datum ist *tag tt–mm–*19*jj*
  Neues Datum eingeben (tt–mm–jj)):_) an. Jetzt kann ein neues Datum im angezeigten Format eingegeben oder das angezeigte Datum kann durch Drücken der Eingabetaste bestätigt werden. Solange der Rechner eingeschaltet bleibt, werden Datum und Uhrzeit von der Systemuhr weitergeführt.

- Das Format des Datums wird vom COUNTRY–Befehl in der CONFIG.SYS–Datei bestimmt (s.a. Kapitel 2.3). Die Tages–, Monats– und Jahresangaben können voneinander durch Punkte (.), Bindestriche (–) oder Schrägstriche (/) getrennt werden.

- Verfügen Sie über einen Commodore PC 40, so braucht bei diesem Rechner das Datum nur ein einziges Mal mit Hilfe des Programms SETUP gesetzt zu werden, da Datum und Uhrzeit mit einer batteriegepufferten Uhr auch dann weitergeführt werden, wenn der Rechner ausgeschaltet ist. Wenn hier jedoch das Datum mit dem DATE–Befehl geändert wird, so gilt diese Änderung nur solange, wie der Rechner nicht ausgeschaltet wird. Die batteriegepufferte Systemuhr kann also nur mit dem Programm SETUP (s. Anhang A im *MS–DOS–Bedienungshandbuch) gestellt werden.*

DEL

Formate: **DEL** [***Laufw***:][***Pfad***]***Datei***[***.Erw***]

oder

ERASE [***Laufw***:][***Pfad***]***Datei***[***.Erw***]

Typ: intern

Bedeutung: Löscht die angegebene(n) Datei(en).

Laufw: Eine gültige Laufwerksbezeichnung nach MS–DOS–Konventionen für die Disk, auf der eine Datei gelöscht werden soll. Wird ***Laufw***: weggelassen, so nimmt MS–DOS das aktuelle Laufwerk an.

Pfad Der Pfad (s.a. Kapitel 4.2.1), auf dem die bezogene Datei zu finden ist. Wird ***Pfad*** weggelassen, so nimmt MS–DOS das aktuelle Verzeichnis an.

Datei[***.Erw***] Der vollständige Name der zu löschenden Datei. Hat die zu löschende Datei eine Namenserweiterung, so muß diese auch angegeben werden. Die Jokerzeichen * und ? für globale Dateinamen sind erlaubt. MS–DOS löscht dann alle Dateien, auf die der angegebene globale Dateinamen zutrifft. Wird der Befehl

```
del *.*
```

eingegeben, so werden **alle** Dateien im aktuellen Verzeichnis gelöscht. Damit es nicht zu ungewollten Datenverlusten kommt, fordert MS–DOS in diesem Fall mit der Anzeige

```
Are you sure (Y/N)?
```

(Sind Sie sicher (J/N)?) eine Bestätigung. Wird die Y–Taste (Yes = Ja) gedrückt, so wird der Befehl ausgeführt. Bei N (No = Nein) nicht.

Beispiele: Mit dem folgenden Befehl wird die Datei TERMINE im aktuellen Verzeichnis der Disk im aktuellen Laufwerk gelöscht.

```
del termine
```

Sollen alle Dateien TERMINE mit beliebiger Namenserweiterung im Unterverzeichnis MARIA auf der Disk in Laufwerk A gelöscht werden, so müssen Sie eingeben:

```
del a:\maria\termine.*
```

Anmerkungen:

- Gelöschte Dateien können nicht wiederhergestellt werden.
- Als Synonym für den Befehl DEL kann auch das Befehlswort ERASE verwendet werden.
- Dateien mit dem Attribut R (nur Lesen) können nicht gelöscht werden. Bei solchen Dateien meldet MS–DOS als Antwort auf einen Löschversuch mit DEL oder ERASE

  ```
  access denied
  ```

  (Zugriff nicht erlaubt).
- Mit DEL oder ERASE können keine Unterverzeichnisse gelöscht werden. Hierzu muß der Befehl RMDIR (s. dort in diesem Kapitel) verwendet werden.

DIR

Format: **DIR** [***Laufw***:][***Pfad***][***Datei***[.***Erw***]] [/**P**][/**W**]

Typ: intern

Bedeutung: Zeigt die Dateieinträge des angegebenen Verzeichnisses an.

Laufw: Eine gültige Laufwerksbezeichnung nach MS–DOS–Konventionen für die Disk, auf der das anzuzeigende Verzeichnis zu finden ist. Wird ***Laufw***: weggelassen, so nimmt MS–DOS das aktuelle Laufwerk an.

Pfad Der Pfad (s.a. Kapitel 4.2.1), für das anzuzeigende Verzeichnis. Wird ***Pfad*** weggelassen, so nimmt MS–DOS das aktuelle Verzeichnis an.

Datei[.***Erw***] Ein Dateiname, ggf. mit Namenserweiterung. Wird er angegeben, so wird nur der Verzeichniseintrag für diese Datei angezeigt. Wird kein Name angegeben, so werden alle Einträge im angegebenen oder aktuellen Verzeichnis angezeigt. Hier sind die Jokerzeichen * und ? für globale Dateinamen erlaubt. Werden sie verwendet, so zeigt der DIR–Befehl die Verzeichniseinträge für alle Dateien, auf die der angegebene globale Dateinamen zutrifft.

/P Wird dieser Schalter angegeben, so zeigt DIR das Verzeichnis bildschirmweise an. Ist der Bildschirm voll, so kann die Anzeige durch Drücken einer beliebigen Taste fortgesetzt werden.

/W Wird dieser Schalter angegeben, so wird das Anzeigeformat geändert. Es werden dann nur die Dateinamen, nicht aber Größe sowie Datum und Uhrzeit der Erstellung oder letzten Änderung angezeigt.

Beispiele: Die vollständige Liste aller Einträge im Basis–Verzeichnis der Disk im aktuellen Laufwerk A erhalten Sie mit

```
dir
```

Die Anzeige sieht dann z.B. so aus:

```
Volume in drive A is BUERO
Directory of A:\

REPORT  TXT    6537      2.09.86   15.30
TERMINE          123     15.09.86    8.25
RECHN      <DIR>          1.10.86   12.36
NOTIZEN TXT    2530     12.10.86    9.15
  4 File(s)   279315 bytes free
```

Dieses Verzeichnis enthält 3 Dateieinträge sowie den Eintrag für ein Unterverzeichnis, der mit <DIR> gekennzeichnet ist.

Die Liste aller Einträge im Unterverzeichnis RECHN dieser Disk erhalten Sie mit dem Befehl

```
dir \rechn
```

und die Liste selbst könnte dann so aussehen:

```
Volume in drive A is BUERO
Directory of A:\RECHN

.          <DIR>          1.10.86   12.36
..         <DIR>         15.08.86   15.30
MUELLER  AUG    730       1.10.86   12.40
MUELLER  SEP   1140       9.10.86    9.12
  2 File(s)   279315 bytes free
```

Für einen einzelnen Eintrag sieht der DIR–Befehl z.B. so aus:

```
dir \rechn\mueller.*
```

und Sie erhalten als Anzeige beispielsweise:

```
Volume in drive A is BUERO
Directory of A:\RECHN

MUELLER  AUG     730      1.10.86   12.40
MUELLER  SEP    1140      9.10.86    9.12
  2 File(s)    279315 bytes free
```

Anmerkungen:

- Die Anzeige enthält zunächst die ggf. vorhandene Diskkennung, den Pfad zum angezeigten Verzeichnis, dann alle Einträge und zum Schluß die Anzahl der Einträge sowie den noch verfügbaren Platz auf der Disk in Bytes. Der verfügbare Platz ist ein ganzzahliges Vielfaches von 1024 Bytes.

- Ein vollständiger Dateieintrag besteht aus dem Dateinamen, der ggf. vorhandenen Namenserweiterung, der Dateigröße in Bytes, dem Datum sowie der Uhrzeit der Dateierzeugung oder letzten –Änderung.

- Das Anzeige–Format von Datum und Uhrzeit hängt von dem Code ab, der mit dem COUNTRY–Befehl in der CONFIG.SYS–Datei (s. Kapitel 2.3) gesetzt wurde.

- Der DIR–Befehl benötigt nicht unbedingt Jokerzeichen zur Auswahl globaler Dateinamen. Deshalb sind folgende Angaben gleichwertig:

Befehl	**Äquivalent**
DIR	DIR *.*
DIR *Datei*	DIR *Datei*.*
DIR .*Erw*	DIR *.*Erw*

DISKCOMP

Format: [*L*:][*BPfad*]**DISKCOMP** [*QLaufw*:[*ZLaufw*:]][/1][/8]

Typ: extern; bei Netzwerkbetrieb nicht erlaubt.

Bedeutung: Vergleicht den Inhalt der Diskette im Quellaufwerk mit der Disk im Ziellaufwerk und zeigt Unterschiede an.

L*:*BPfad Laufwerk und Suchpfad für die DISKCOMP-Befehlsdatei.

***QLaufw*:** Eine gültige Laufwerksbezeichnung nach MS–DOS–Konventionen (A oder B) für die Quelldiskette.

***ZLaufw*:** Eine gültige Laufwerksbezeichnung nach MS–DOS–Konventionen (A oder B) für die Zieldiskette, die mit der Quelldiskette verglichen werden soll.

Wird nur ein Laufwerk angegeben, benutzt MS–DOS das aktuelle Laufwerk als Ziellaufwerk. Sind beide Laufwerksangaben gleich, so führt MS–DOS den Vergleich mit einem Laufwerk durch und fordert rechtzeitig zum jeweiligen Diskettenwechsel auf. Es dürfen nur Diskettenlaufwerke angegeben werden. Andernfalls zeigt MS–DOS eine Fehlermeldung an.

/1 Wird dieser Schalter angegeben, so wird nur eine Seite der Disketten gelesen und verglichen, auch wenn es sich um doppelseitige Disketten handelt.

/8 Wird dieser Schalter angegeben, so werden nur die ersten 8 Sektoren jeder Spur gelesen, auch wenn die Quelldiskette 9 oder 15 Sektoren/Spur enthält.

Beispiele: Wenn die MS–DOS–Systemdateien auf der Festplatte (Laufwerk C) gespeichert sind, Laufwerk A das aktuelle Laufwerk ist und zwei doppelseitige Disketten mit 9

Sektoren/Spur miteinander verglichen werden sollen, müssen Sie folgenden Befehl eingeben:

```
c:\diskcomp
```

Sollen zwei Laufwerke für den Vergleich zweier einseitiger Disketten mit 8 Sektoren/Spur benutzt werden und ist kein Festplattenlaufwerk verfügbar, so muß zunächst die MS–DOS–Systemdiskette eingelegt und dann eingegeben werden:

```
diskcomp a: b:/1/8
```

Anmerkungen:

- DISKCOMP führt einen Spur–zu–Spur–Vergleich bei beiden Disketten durch und bestimmt dabei automatisch die Anzahl der Seiten und der Sektoren/Spur. Wenn die Zieldiskette nicht vom gleichen Typ wie die Quelldiskette ist, wird die Fehlermeldung

  ```
  Drive types or diskette types not compatible
  ```

  (Laufwerk– oder Diskettentyp nicht kompatibel) angezeigt.

- Hat der Vergleich keine Unterschiede ergeben, so zeigt MS–DOS die Meldung

  ```
  Compare OK
  ```

  (Vergleich OK) an. Bei fehlender Übereinstimmung wird eine Fehlermeldung zusammen mit der Spur– und Seiten–Nummer, bei der der Fehler gefunden wurde, angezeigt.

- Nach Beendigung des Vergleiches erscheint die Systemanfrage

  ```
  Compare another Diskette (Y/N)?
  ```

  (Eine weitere Diskette vergleichen (J/N)?). Wird mit Y (Yes = Ja) geantwortet, so fordert MS–DOS zum

Einlegen der richtigen Diskette auf. Bei N (No = Nein) wird DISKCOMP beendet und die Bereitschaftsanzeige erscheint wieder. War A das aktuelle Laufwerk, so fordert MS–DOS mit der Meldung

```
Insert disk with COMMAND.COM in drive
A and strike any key when ready
```

(Diskette mit COMMAND.COM in Laufwerk A einlegen. Wenn bereit, beliebige Tasten drücken) die Systemdisk an, damit der transiente Teil des Befehlsprozessors wieder geladen werden kann.

- DISKCOMP arbeitet nicht bei Netzwerkbetrieb, oder wenn Laufwerke mit den Befehlen ASSIGN, JOIN oder SUBST anders zugewiesen wurden.

- Wenn Sie eine Diskette mit einer Kopie, die Sie mit dem COPY–Befehl erstellt haben, vergleichen, kann es zu Fehlermeldungen kommen, auch wenn die Dateien identisch sind. COPY dupliziert zwar die Informationen, schreibt sie aber nicht notwendigerweise an dieselbe Stelle der Kopie, an der sie im Original gestanden haben. Das gilt insbesondere für stark fragmentierte Originaldateien. In einem solchen Fall sollte ein Dateivergleich mit dem FC–Programm (s. Kapitel 9) durchgeführt werden.

- Der DISKCOMP–Befehl erzeugt bei seiner Beendigung die folgenden Fehlergrad–Codes, die mit dem Stapelverarbeitungsbefehl IF ERRORLEVEL (s. dort in Kapitel 6.3.2) verarbeitet werden können, wenn DISKCOMP in einer Stapeldatei verwendet wird.

  0 Normale Beendigung; kein Fehler.
  1 Wegen unterschiedlicher Disketten kein Vergleich möglich.
  2 Abbruch durch den Benutzer mit Ctrl–Break oder Ctrl–C.
  3 Wegen nicht behebbarem Lesefehler kein Vergleich möglich.
  4 Wegen unzureichendem Speicherplatz, ungültiger Laufwerksangabe oder Syntaxfehler in der Befehlszeile kein Vergleich möglich.

DISKCOPY

Format: [*L*:][*BPfad*]**DISKCOPY** [*QLaufw*:[*ZLaufw*:]][/1]

Typ: extern; bei Netzwerkbetrieb nicht erlaubt.

Bedeutung: Dupliziert den Inhalt der Diskette im Quellaufwerk auf die Diskette im Ziellaufwerk.

L*:*BPfad Laufwerk und Suchpfad für die DISKCOPY-Befehlsdatei.

***QLaufw*:** Eine gültige Laufwerksbezeichnung nach MS–DOS–Konventionen (A oder B) für die Quelldiskette.

***ZLaufw*:** Eine gültige Laufwerksbezeichnung nach MS–DOS–Konventionen (A oder B) für die Zieldiskette, auf die die Quelldiskette dupliziert werden soll.

Wird nur ein Laufwerk angegeben, benutzt MS–DOS das aktuelle Laufwerk als Ziellaufwerk. Sind beide Laufwerksangaben gleich, so führt MS–DOS die Duplizierung mit einem Laufwerk durch und fordert rechtzeitig zum jeweiligen Diskettenwechsel auf. Es dürfen nur Diskettenlaufwerke angegeben werden. Andernfalls zeigt MS–DOS eine Fehlermeldung an.

/1 Wird dieser Schalter angegeben, so wird nur eine Seite der Quelldiskette gelesen, auch wenn es sich um doppelseitige Disketten handelt.

Beispiele: Wenn die MS–DOS–Systemdateien auf der Festplatte (Laufwerk C) gespeichert sind und Laufwerk A das aktuelle Laufwerk ist, müssen Sie folgenden Befehl eingeben:

```
c:\diskcopy a:
```

Sollen zwei Laufwerke für die Duplizierung zweier Disketten benutzt werden, soll nur eine Seite dupliziert werden und ist kein Festplattenlaufwerk verfügbar, so

muß zunächst die MS–DOS–Systemdiskette eingelegt und dann eingegeben werden:

```
diskcopy a: b:/1
```

Anmerkungen:

- DISKCOPY formatiert während des Duplizierungsvorganges die Zieldiskette automatisch und erkennt selbsttätig Anzahl der Seiten sowie Anzahl der Sektoren/Spur bei der Quelldiskette. In jedem Fall wird der Inhalt der Zieldiskette vor der Duplizierung durch DISKCOPY zerstört.

- Nach Beendigung der Duplizierung erscheint die Systemanfrage

  ```
  Copy another Diskette (Y/N)?
  ```

  (Eine weitere Diskette duplizieren (J/N)?). Wird mit Y (Yes = Ja) geantwortet, so fordert MS–DOS zum Einlegen der Quell–Diskette auf. Bei N (No = Nein) wird DISKCOPY beendet und die Bereitschaftsanzeige erscheint wieder. War A das aktuelle Laufwerk, so fordert MS–DOS mit der Meldung

  ```
  Insert disk with COMMAND.COM in drive
  A and strike any key when ready
  ```

  (Diskette mit COMMAND.COM in Laufwerk A einlegen. Wenn bereit, beliebige Taste drücken) die Systemdisk an, damit der transiente Teil des Befehlsprozessors wieder geladen werden kann.

- DISKCOPY arbeitet nicht bei Netzwerkbetrieb, oder wenn Laufwerke mit den Befehlen ASSIGN, JOIN oder SUBST anders zugewiesen wurden.

- Wenn Ihr Computer nur mit einem Diskettenlaufwerk ausgerüstet ist, können Sie als Ziellaufwerk dennoch Laufwerk B angeben. MS–DOS nimmt dann automatisch dafür Laufwerk A.

- Da MS–DOS Disk–Speicherplatz nicht sequentiell zuweist, sind häufig erzeugte, fortgeschriebene und gelöschte Dateien stark fragmentiert. Aus diesem Grund wird der erste von DISKCOPY gefundene freie Sektor der nächste zugewiesene, unabhängig von seiner Position auf der Diskette. Bei einer fragmentierten Disk verlangsamt sich der Zugriff zum Schreiben oder Lesen. In solchen Fällen sind die beiden Befehle COPY und XCOPY (s. dort in diesem Kapitel) zum Erstellen von Sicherungskopien besser geeignet.

- Der DISKCOPY–Befehl erzeugt bei seiner Beendigung die folgenden Fehlergrad–Codes, die mit dem Stapelverarbeitungsbefehl IF ERRORLEVEL (s. dort in Kapitel 6.3.2) verarbeitet werden können, wenn DISKCOPY in einer Stapeldatei verwendet wird.

  0 Normale Beendigung; kein Fehler.
  1 Nicht–fataler Schreib–/Lese–Fehler beim Duplizieren.
  2 Abbruch durch den Benutzer mit Ctrl–Break oder Ctrl–C.
  3 Wegen nicht behebbarem Schreib–/Lese–Fehler keine Duplizierung möglich.
  4 Wegen unzureichendem Speicherplatz, ungültiger Laufwerksangabe oder Syntaxfehler in der Befehlszeile keine Duplizierung möglich.

ERASE

Siehe DEL. ERASE ist ein Synonym für den DEL–Befehl.

EXE2BIN

Format: [*L*:][*BPfad*]**EXE2BIN** [*Laufw*:][*Pfad*] *Datei*[*.Erw*] [*Laufw*:][*Pfad*] [*Datei*][*.Erw*]]

Typ: extern

Bedeutung: Wandelt .EXE–Dateien in das .COM oder .BIN–Format.

L*:*BPfad Laufwerk und Suchpfad für die EXE2BIN–Befehlsdatei.

Laufw*:*PfadDatei.Erw Die vollständigen Angabe für die Quelldatei (d.i. die erste Dateiangabe für die umzuwandelnde Datei) und die Zieldatei (d.i. die zweite Dateiangabe für die .COM– oder .BIN–Datei). Wird das Quellaufwerk weggelassen, so nimmt MS–DOS das aktuelle Laufwerk an. Wird das Ziellaufwerk weggelassen, so nimmt MS–DOS das Quellaufwerk an. Wird der Pfad für Quell– und/oder Zieldatei weggelassen, so nimmt MS–DOS das aktuelle Verzeichnis an. Wird der Zieldateiname weggelassen, so nimmt MS–DOS den Quelldateinamen. Wird die Namenserweiterung bei der Quelldatei weggelassen, so nimmt MS–DOS .EXE an, beim Zieldateinamen .BIN.

Beispiel: Um die Datei MEINPROG.EXE in die Datei MEINPROG .COM im aktuellen Verzeichnis der Disk im aktuellen Laufwerk umzuwandeln, müssen Sie eingeben:

```
exe2bin meinprog .com
```

Anmerkungen:

- Bei der Anwendung des EXE2BIN–Befehls sind einige Einschränkungen zu beachten:

  Die Eingabedatei muß im gültigen .EXE–Format durch das Binde–Programm LINK (s.a. Kapitel 10) erstellt worden sein.

Der residente Teil der zu wandelnden Datei (das sind der Code selbst sowie ggf. vorhandene Daten) darf nicht mehr als maximal 64 KBytes umfassen.

Das .EXE–Format darf **kein** Stapelsegment enthalten.

Ist eine dieser Einschränkungen doch gegeben, so wird EXE2BIN mit der Meldung

```
File cannot be konverted
```

(Datei kann nicht gewandelt werden) abgebrochen.

- Abhängig davon, ob in der .EXE–Datei **CS:IP** (Code–Segment:Befehlszähler) spezifiziert wurde, sind mit EXE2BIN zwei verschiedene Umwandlungen möglich:

  1. Ist CS:IP in der .EXE–Datei nicht gesetzt, nimmt EXE2BIN eine reine Binär–Wandlung an. Enthält die .EXE–Programmdatei Befehle für Segmentverschiebungen (sogenannte Segmentfixierungen), so wird der Benutzer zur Eingabe eines entsprechenden Fixierwertes aufgefordert. Dieser Wert definiert das absolute Segment, in das das Programm geladen werden muß. Das geladene Programm ist nur dann ausführbar, wenn es an die absolute Speicheradresse geladen wird, die durch die jeweilige Anwendung festgelegt wird. Ein solches Programm kann vom Befehlsprozessor nicht geladen werden.

  2. Wird CS:IP auf 0000:100H gesetzt, nimmt EXE2BIN an, daß das betreffende Programm als .COM–Datei ausgeführt werden soll, bei der der Adreßzeiger vom Assembler–Pseudobefehl ORG auf die Adresse 100H gesetzt wurde. In diesem Fall sind keine Segmentfixierungen erlaubt, da .COM–Dateien frei segment–verschiebbar (relokatierbar) sein müssen. Nach der Umwandlung kann die Ergebnisdatei mit einer

.COM–Namenserweiterung umbenannt werden. Der MS–DOS–Befehlsprozessor kann eine solche Datei dann genauso ausführen wie die anderen auf der MS–DOS–Systemdisk gespeicherten .COM–Dateien.

EXIT

Format: **EXIT**

Typ: intern

Bedeutung: Wechselt vom aktuellen COMMAND.COM–Befehlsprozessor zum nächst höheren, falls vorhanden.

Anmerkungen:

- Wenn Sie mit dem COMMAND–Befehl (s. dort in diesem Kapitel) einen neuen Befehlsprozessor gestartet haben, können Sie diesen mit dem Befehl EXIT wieder verlassen und den vorhergehenden, auf der nächst höheren Ebene liegenden wieder aktivieren.

- Ebensogut können Sie auch ein Anwenderprogramm, das unter MS–DOS läuft, zum Befehlsprozessor verlassen und dann mit dem EXIT–Befehl wieder fortsetzen.

FDISK

Format: **FDISK**

Typ: extern; bei Netzwerkbetrieb nicht erlaubt.

Bedeutung: Bereitet die Festplatte für MS–DOS vor.

Anmerkungen:

- Da die Festplatten–Konfigurierung eine wichtige und unabdingbare Maßnahme ist, wenn eine neue Festplatte für die Arbeit mit MS–DOS verwendet werden soll, ist dafür ein eigenes Kapitel in diesem Handbuch vorgesehen (s. Kapitel 3).

FIND

Format: [*L*:][*BPfad*]**FIND** [/V][/C][/N] "*Zeichenkette*" [[*Laufw*:][*Pfad*]*Datei*[.*Erw*]. . .]

Typ: extern

Bedeutung: FIND ist ein Filter–Befehl (s.a. Kapitel 5.5.2) und durchsucht alle angegebenen Dateien nach der angegebenen Zeichenkette. Jede Zeile, die die Zeichenkette enthält, wird angezeigt.

L*:*BPfad Laufwerk und Suchpfad für die FIND–Befehlsdatei.

/V Wird dieser Schalter angegeben, so werden alle Zeilen, die die angegebene Zeichenkette **nicht** enthalten, angezeigt.

/C Wird dieser Schalter angegeben, so wird nur die Anzahl der Zeilen, die die angegebene Zeichenkette enthalten, angezeigt.

/N Wird dieser Schalter angegeben, so wird jeder angezeigten Zeile eine Zeilennummer vorangestellt.

"*Zeichenkette*" Eine Folge aus beliebigen Zeichen, die in Anführungsstriche (") eingeschlossen sein muß, und nach der die angegebene(n) Datei(en) durchsucht werden soll(en).

***Laufw*:*PfadDatei.Erw*. . .** Die vollständige(n) Angabe(n) für alle zu durchsuchenden Dateien. Wird FIND ohne Dateiangabe verwendet, holt FIND seine Eingabe vom Standard–Eingabegerät (üblicherweise die Tastatur) und zeigt jede Zeile an, die die angegebene Zeichenkette enthält. Kommt die Eingabe von der Tastatur, so wird FIND entweder durch aufeinanderfolgendes Drükken der F6 (Ctrl–Z)– und der Eingabetaste oder durch Drücken der Tastenkombination Ctrl–Break oder Ctrl–C beendet.

Beispiele:

Wenn Sie alle Zeilen in den drei Dateien TEXT1.TXT, TEXT2.TXT und TEXT3.TXT im aktuellen Verzeichnis der Disk im aktuellen Laufwerk angezeigt bekommen wollen, in denen das Wort **Gartenzaun** vorkommt, so müssen Sie eingeben:

```
find "Gartenzaun" text1.txt text2.txt
text3.txt
```

Wenn Sie wissen wollen, wieviele Dateieinträge im Basisverzeichnis der Disk in Laufwerk A **nicht** die Zeichenfolge **COM** enthalten, müssen Sie eingeben:

```
dir a:\|find /v"COM"
```

Wenn Sie wissen wollen, wieviele Unterverzeichnisse im Basisverzeichnis der Festplatte in Laufwerk C enthalten sind, müssen Sie eingeben:

```
dir c:\|find /c"<DIR>"
```

Anmerkungen:

- Wird nach einer Zeichenkette gesucht, die selbst Anführungszeichen (") enthält, so muß die bei FIND angegebene Zeichenkette an der Stelle, an der die gesuchten Anführungszeichen stehen, doppelte Anführungszeichen ("") enthalten. Wird also z.B. nach dem Text **der "Commodore PC-AT" ist ein schneller Computer** gesucht, so muß diese Zeichenkette als

  ```
  find "der ""Commodore PC-40"" ist ein
  schneller Computer"
  ```

  angegeben werden.

- Bei der Angabe der Dateinamen und Namenserweiterungen sind **keine** JokerZeichen (* und/oder ?) für globale Dateinamen erlaubt.

FORMAT

Formate: **[*L*:][*BPfad*]FORMAT [*Lauf*:][/1][/4][/8] [/N:*xx*][/T:*yy*][/V][/S]**

oder

[*L*:][*BPfad*]FORMAT [*Laufw*:][/1][/B] [/N:*xx*][/T:*yy*]

Typ: extern; bei Netzwerkbetrieb nicht erlaubt.

Bedeutung: Formatiert die Diskette oder Festplatte im angegebenen Laufwerk, damit MS–DOS dort Dateien erzeugen kann.

L*:*Pfad Laufwerk und Suchpfad für die FORMAT–Befehlsdatei.

***Laufw*:** Eine gültige Laufwerksbezeichnung nach MS–DOS–Konventionen für die zu formatierende Disk. Wird ***Laufw*:** nicht angegeben, so meldet MS–DOS

```
Drive letter must be specified
```

(Laufwerksbuchstabe muß angegeben werden).

/1 Wird dieser Schalter angegeben, so wird die Diskette nur einseitig formatiert, gleichgültig, ob es sich um ein doppelseitiges Laufwerk oder eine doppelseitige Diskette handelt. Dieser Schalter erlaubt die Formatierung von Disketten für einseitige Laufwerke.

/4 Wird dieser Schalter angegeben, so wird eine doppelseitige Diskette (double density (360 KBytes)) in einem 1,2–MBytes–Laufwerk (high density) formatiert. Beachten Sie jedoch, daß Sie solche Disketten in einem 360–KBytes–Laufwerk nicht absolut zuverlässig lesen können (s.a. *MS–DOS–Bedienungshandbuch*, Kapitel 1.6).

/8 Wird dieser Schalter angegeben, so wird die Disk mit 8 Sektoren pro Spur formatiert. Wird er nicht angegeben,

so formatiert der FORMAT–Befehl standardmäßig mit 9 oder 15 Sektoren pro Spur, je nach verwendetem Laufwerkstyp (double densitiy oder high density). Beachten Sie hier, daß MS–DOS tatsächlich immer mit 9 bzw. 15 Sektoren pro Spur formatiert, jedoch in diesem Fall nur 8 Sektoren **nutzt**.

/B Wird dieser Schalter angegeben, so wird die Disk ebenfalls mit 8 Sektoren pro Spur formatiert. Zusätzlich wird für die beiden verborgenen Systemdateien MSDOS.SYS und IO.SYS auf der Disk Platz reserviert, diese werden jedoch nicht übertragen. Auf eine solche Disk kann mit dem SYS–Befehl jeder früheren Version von MS–DOS diese auf die Disk übertragen werden. Ohne diesen Schalter können Sie nur System–Disketten für die MS–DOS–Version 3.20 erzeugen. Die /S– und /V–Schalter (s.u.) können nicht zusammen mit dem /B–Schalter verwendet werden.

/N:*xx* Mit diesem Schalter kann festgelegt werden, mit wieviel (*xx*) Sektoren pro Spur der FORMAT–Befehl Disketten formatieren soll.

/T:*yy* Dieser Schalter legt fest, wieviele (*yy*) Spuren der FORMAT–Befehl auf einer Diskette erzeugen soll.

/V Wird dieser Schalter angegegeben, so fordert der FORMAT–Befehl nach dem Abschluß einer erfolgreichen Formatierung die Eingabe einer 11 Zeichen langen Disk-Kennung, die als Kennsatz auf die Disk übertragen wird. Damit hat man die Möglichkeit, wichtigen Disketten Namen zu geben (s.a. LABEL–Befehl in diesem Kapitel).

/S Wird dieser Schalter angegeben, so muß er als letzter Schalter in der Befehlszeile angegeben werden. MS–DOS überträgt dann nach erfolgreicher Formatierung die drei Systemdateien in dieser Reihenfolge auf die Disk:

1. IO.SYS (verborgene Datei)
2. MSDOS.SYS (verborgene Datei)
3. COMMAND.COM

Befindet sich dazu das Betriebssystem nicht auf der Disk im aktuellen Laufwerk, so fordert MS–DOS dazu auf, die Systemdisk ins aktuelle Laufwerk (oder in Laufwerk A, wenn das aktuelle Laufwerk die Festplatte ist) einzulegen.

Beispiele:

Um eine Diskette im Laufwerk A zu formatieren und die Systemdateien darauf zu übertragen, müssen Sie eingeben:

```
format a: /s
```

Soll diese Diskette außerdem eine Kennung erhalten, geben Sie ein:

```
format a: /s/v
```

Am Ende der Formatierung fordert MS–DOS dann die Eingabe einer bis zu 11 Zeichen langen Kennung, die als Disk–Namen auf die Diskette geschrieben wird.

Anmerkungen:

- Durch die Formatierung werden alle ggf. vorher auf der angegebenen Disk gespeicherten Daten zerstört. Deshalb ist insbesondere bei der Formatierung von Festplatten große Vorsicht geboten.

- Der FORMAT–Befehl initialisiert auf der formatierten Disk das Basisverzeichnis, die Dateizuordnungstabelle und den Systemlader. Ehe eine Disk mit MS–DOS benutzt werden kann, muß sie formatiert werden.

- Zusammen mit dem FORMAT–Befehl muß das Laufwerk angegeben werden, in dem die zu formatierende Disk eingelegt ist. FORMAT benutzt dann den Laufwerkstyp, um das Standard–Formatierformat festzulegen.

- Wenn Sie eine Festplatte formatieren, fordert MS–DOS Sie zunächst mit der Anzeige

```
Enter current Volume Label for drive x:
```

(Geben Sie den aktuellen Disk–Kennsatz für Laufwerk *x* ein) auf, den ggf. bereits vorhandenen Disk-Kennsatz einzugeben. Verfügt die Festplatte über keinen Kennsatz, drücken Sie einfach die Eingabetaste. Stimmen eingegebener und vorhandener Kennsatz nicht überein, bricht MS–DOS die Formatierung mit der Meldung

```
Invalid Volume ID Format Failure
```

(Ungültige Disk–Kennung, Formatierungs–Fehler) ab, andernfalls werden Sie mit der Anzeige

```
WARNING, ALL DATA ON NON-REMOVABLE
DISK
DRIVE x WILL BE LOST!
Proceed with Format (Y/N)?
```

(WARNUNG, SÄMTLICHE DATEN AUF DEM FESTPLATTENLAUFWERK *x* WERDEN ZERSTÖRT! Mit der Formatierung fortfahren (J/N)?) aufgefordert, die Formatierung zu bestätigen. Drücken Sie jetzt die Y–Taste (Yes = Ja) und dann die Eingabetaste, so wird die Disk formatiert. Bei N (No = Nein) und der Eingabetaste wird der FORMAT–Befehl abgebrochen.

- Nach beendeter Formatierung erstellt FORMAT einen Statusbericht über die formatierte Disk, der neben dem verfügbaren Speicherplatz auf der Disk defekte und markierte Sektoren, den vom System belegten Speicherplatz (nur, wenn der /S–Schalter verwendet wurde) sowie den für den Anwender verfügbaren Speicherplatz enthält.

- Die folgende Tabelle zeigt, welche der Schalter für welchen Disk–Typ zur Formatierung erlaubt sind:

Disk–Typ	**erlaubte Schalter**
160/320 KBytes	/1 /4 /8 /B /N /T /V /S
320/360 KBytes	/1 /4 /8 /B /N /T /V /S
720 KBytes	/N /T /V /S
Festplatte	/V /S

- Der FORMAT–Befehl ignoriert Laufwerkszuweisungen mit dem ASSIGN–Befehl. Sie sollten außerdem darauf achten, daß Sie alle Zuweisungen, die Sie mit den Befehlen JOIN oder SUBST (s. dort in diesem Kapitel) vorgenommen haben, erst aufheben, bevor sie eine Disk in einem betroffenen Laufwerk formatieren.

- Bei Netzwerkbetrieb ist die Benutzung des FORMAT–Befehls nicht erlaubt.

- Wurde bei einer Festplatte die Größe der DOS–Partition mit dem FDISK–Befehl (s.a. Kapitel 3.2.2) geändert, so muß diese Festplatte mit FORMAT neu formatiert werden.

- Der FORMAT–Befehl erzeugt bei seiner Beendigung die folgenden Fehlergrad–Codes, die mit dem Stapelverarbeitungsbefehl IF ERRORLEVEL (s. dort in Kapitel 6.3.2) verarbeitet werden können, wenn FORMAT in einer Stapeldatei verwendet wird.

  0 Normale Beendigung; kein Fehler.
  3 Abbruch durch den Benutzer mit Ctrl–Break oder Ctrl–C.
  4 Fataler Fehler (jeder andere Fehler als 0, 3 oder 5).
  5 Bei der Systemanfrage

  ```
  Proceed with Format (Y/N)?
  ```

  wurde mit N geantwortet.

GRAFTABL

Format: [*L*:][*BPfad*]**GRAPHTABL**

Typ: extern

Bedeutung: Lädt eine zusätzliche Zeichentabelle für die Verwendung in Verbindung mit einem Farb–/Grafik–Adapter.

L:Pfad Laufwerk und Suchpfad für die GRAFTABL–Befehlsdatei.

Anmerkungen:

- GRAFTABL lädt eine Tabelle mit den ASCII–Zeichen der Codes 128 bis 255 dez. in den Speicher. Wenn Ihr Computer über einen Farb–/Grafik–Adapter verfügt, können Sie dann fremdsprachliche Zeichen und grafische Symbole anzeigen, wenn Sie sich im Grafik-Modus der Anzeige befinden.

- GRAPHTABL meldet das erfolgreiche Laden der Tabelle mit

  ```
  Graphic characters loaded
  ```

  (Grafische Zeichen geladen). Die Tabelle kann nach dem Starten von MS–DOS jedoch nur ein einziges Mal geladen werden. Wird wiederholtes Laden versucht, erhält man die Anzeige

  ```
  Graphic characters already loaded
  ```

  (Grafische Zeichen bereits geladen).

- GRAFTABL erhöht den Speicherbedarf des residenten Teils von MS–DOS.

GRAPHICS

Format: [*L*:][*BPfad*]**GRAPHICS** [*Drucker*][/**R**][/**B**]

Typ: extern

Bedeutung: Lädt die Grafik–Druck–Unterstützung zur Ausgabe von grafischen Bildschirmdarstellungen auf einem IBM-kompatiblen Matrixdrucker, wenn ein Farb–/Grafik–Adapter benutzt wird.

L*:*Pfad Laufwerk und Suchpfad für die GRAPHICS–Befehlsdatei.

Drucker Hier können folgende Parameterkonstanten angegeben werden:

COLOR1 erlaubt den Ausdruck auf einem Farb–Matrixdrucker mit schwarzem Farbband.

COLOR4 erlaubt den Ausdruck auf einem Farb–Matrixdrucker mit RGB–Farbband (rot, grün, blau, schwarz).

COLOR8 erlaubt den Ausdruck auf einem Farb–Matrixdrucker mit 4–Farb–Band (kobaltblau, violett, gelb, schwarz).

COMPACT erlaubt den Ausdruck auf einem Kompakt-Drucker.

GRAPHICS erlaubt den Ausdruck auf einem normalen Matrixdrucker.

Wird ***Drucker*** weggelassen, so gilt GRAPHICS als Standard–Parameter.

/R Wird dieser Schalter angegeben, so wird schwarz als schwarz und weiß als weiß ausgedruckt, das Druckbild wird also so erzeugt, wie es auf dem Bildschirm erscheint. Wird **/R** nicht angegeben, so wird schwarz als

weiß und weiß als schwarz ausgedruckt, das Druckbild farblich also invertiert.

/B Wird dieser Schalter angegeben (nur bei COLOR4– und COLOR8–Druckern erlaubt), so wird die Hintergrundfarbe mit ausgedruckt, andernfalls nicht.

Beispiel:

Sollen grafische Bildschirmdarstellungen auf einem normalen IBM–kompatiblen Matrixdrucker so ausgedruckt werden, wie sie auf dem Bildschirm erscheinen, müssen Sie zunächst

```
graphics /r
```

eingeben. Zur Ausgabe des Bildschirminhaltes brauchen Sie dann nur noch bei niedergehaltener Umschalt-Taste die PrtSc–Taste (Print Screen = drucke Bildschirm) zu betättigen.

Anmerkungen:

- Zum Ausdruck des Bildschirminhaltes muß die PrtSc–Taste bei niedergehaltener Umschalttaste betätigt werden. Befindet sich der Bildschirm im Textmodus, so wird der Bildschirminhalt unverzüglich ausgedruckt (in ca. 10 Sekunden).

- Befindet sich der Bildschirm im 320x200–Farbgrafik–Modus und wurde der Druckertyp COLOR1 oder GRAPHICS angegeben, so wird der Bildschirminhalt in bis zu vier Grautönen ausgedruckt.

- Befindet sich der Bildschirm im 640x200–Farbgrafikmodus, so wird der Bildschirminhalt seitenverkehrt ausgedruckt, d.h. die rechte obere Bildschirmecke wird auf der linken oberen Papierecke ausgedruckt.

- Der Ausdruck eines grafischen Bildschirms kann je nach verwendetem Druckertyp bis zu 3 Minuten dauern.

- GRAPHICS erhöht den Speicherbedarf des residenten Teils von MS–DOS.

JOIN

Formate: [*L*:][*BPfad*]**JOIN**

oder

[*L*:][*BPfad*]**JOIN** ***Laufw*:*Laufw*:*Verzeichnis**

oder

[*L*:][*BPfad*]**JOIN** ***Laufw*: /D**

Typ: extern

Bedeutung: Verknüpft ein angegebenes Laufwerk mit dem angegebenen Verzeichnis eines anderen Laufwerks zur Bildung einer einzigen Verzeichnisstruktur. Mit Hilfe des JOIN-Befehls ist es nicht mehr erforderlich, physikalische Laufwerke mit eigenen Laufwerksbuchstaben zu benennen. Statt dessen kann auf alle Verzeichnisse auf der Disk in einem Laufwerk über einen einzigen Pfad zugegriffen werden.

L*:*BPfad Laufwerk und Suchpfad für die JOIN–Befehlsdatei.

***Laufw*:** Eine gültige Laufwerksbezeichnung nach MS–DOS–Konventionen für das Laufwerk, das mit dem Verzeichnis des zweiten angegeben Laufwerks (s.u.) verknüpft werden soll.

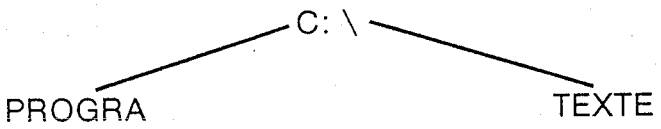
Laufw*:*Verzeichnis Laufwerk– und Verzeichnis–Angabe für das mit dem ersten angegebenen Laufwerk (s.o.) zu verknüpfende Verzeichnis. Dieses Verzeichnis muß im Basisverzeichnis der Disk im angegebenen Laufwerk enthalten sein. Existiert das Verzeichnis dort noch nicht, so wird es angelegt. Existiert es bereits, so muß es leer sein, darf also nur die beiden .– und . .–Einträge enthalten.

***Laufw*: /D** Wird der Schalter /D zusammen mit einem Laufwerk angegeben, so wird die Verknüpfung dieses Laufwerkes wieder aufgenoben.

Wird der JOIN–Befehl ohne Parameter verwendet, so werden alle Laufwerke und Verzeichnisse mit ihren Zugriffspfaden angezeigt, die gegenwärtig verküpft sind.

Beispiele:

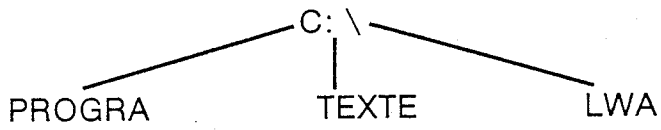
Bei den im folgenden angegebenen Beispielen sei B das aktuelle Laufwerk und das Basisverzeichnis das aktuelle Verzeichnis der Festplatte in Laufwerk C mit der folgenden Struktur:



Das Laufwerk A kann jetzt mit einem Pfad C:\LWA verknüpft werden, indem eingegeben wird:

```
B>join a: c:\lwa
```

Damit ändert sich die Verzeichnisstruktur der Festplatte wie folgt:



Das Verzeichnis LWA wurde, da es vorher noch nicht vorhanden war, im Basisverzeichnis der Festplatte erzeugt.

Jetzt kann das Verzeichnis der Festplatte durch Eingabe von

```
dir c:
```

angezeigt werden. Man erhält

```
Volume in drive C is COMMOAT
Directory of C:\

PROGRA   <DIR>          1.10.86   12.12
TEXTE    <DIR>          4.10.86   15.01
LWA      <DIR>          9.10.86    8.15
 3 File(s)  20279325 bytes free
```

Wenn jetzt eingegeben wird

```
dir a:
```

so erhält man die Anzeige

```
Invalid drive specification
```

(Ungültige Laufwerksangabe), weil auf ein verknüpftes Laufwerk nicht mehr zugegriffen werden kann.

Wird dagegen

```
dir c:\lwa
```

eingegeben, so wird das Verzeichnis der Diskette in Laufwerk A angezeigt. Die Angabe über den noch verfügbaren Speicherplatz bezieht sich allerdings auf Laufwerk C.

Wenn nur die aktuell erzeugten Verknüpfungen angezeigt werden sollen, wird der JOIN–Befehl ohne Parameter benutzt, also:

```
join
```

Dann wird unter den oben angegebenen Voraussetzungen angezeigt:

```
A: => C:\LWA
```

Soll eine bestehende Verknüpfung wieder aufgehoben werden, wird eingegeben:

```
join a: /d
```

Danach kann wieder normal auf Laufwerk A zugegriffen werden.

Anmerkungen:

- Wenn versucht wird, ein Laufwerk mit einem nicht leeren Verzeichnis zu verknüpfen, erhält man die Anzeige

  ```
  Directory not empty
  ```

  (Verzeichnis nicht leer).

- Wenn der angegebene Zugriffspfad mit dem aktuellen Verzeichnis identisch ist, erhält man die Anzeige

  ```
  Invalid parameter
  ```

  (Ungültiger Parameter).

- Im Zugriffspfad darf nicht das Basisverzeichnis angegeben werden.

- Sobald ein Laufwerk mit einem Verzeichnis eines anderen Laufwerkes verknüpft ist, kann auf ersteres nicht mehr zugegriffen werden. Vielmehr muß dann das verknüpfte Verzeichnis angegeben werden. Deshalb kann das aktuelle Laufwerk mit JOIN nicht mit einem anderen Verzeichnis verknüft werden, weil sonst automatisch ein ungültiges Laufwerk entstehen würde.

KEYB*xx*

Format: [*L*:][*BPfad*]KEYB*xx*

Typ: extern

Bedeutung: Lädt die Unterstützung für eine landesspezifische Tastatur. Dieses Programm ersetzt das im BIOS–ROM gespeicherte Standard–Tastaturprogramm für die amerikanische Tastatur.

L*:*BPfad Laufwerk und Suchpfad für die KEYB*xx*–Befehlsdatei.

xx Eine Zwei–Buchstaben–Kennung für das landespezifische Tastaturprogramm. Es kann neben dem residenten amerikanischen Tastaturprogramm zwischen fünf weiteren durch folgende Buchstabenkennungen gewählt werden:

xx	**Land**	**Befehl**
FR	Frankreich	KEYBFR
GR	Deutschland	KEYBGR
IT	Italien	KEYBIT
SP	Spanien	KEYBSP
UK	Großbritannien	KEYBUK

Beispiel: Um das deutsche Tastaturprogramm resident in den Speicher zu laden, sollte der Befehl KEYBGR in die AUTOEXEC.BAT–Datei eingetragen werden, damit dieser Vorgang jedesmal, wenn MS–DOS gestartet wird, automatisch ausgeführt wird.

Anmerkungen:

- Nach dem Start von MS–DOS sollte grundsätzlich nur ein länderspezifisches Tastaturprogramm geladen werden.

- Je nach Landescode vergrößert das Tastaturprogramm den Speicherbedarf des residenten Teils von MS–DOS um einen unterschiedlichen Betrag.

- Durch gleichzeitiges Drücken der drei Tasten Ctrl, Alt und F1 kann vom nationalen Tastaturprogramm KEYB*xx* zum amerikanischen Standard–Tastaturprogramm umgeschaltet werden. Gleichzeitiges Drücken der drei Tasten Ctrl, Alt und F2 schaltet wieder zum nationalen Tastaturprogramm zurück.

LABEL

Format: [*L*:][*BPfad*]**LABEL** [*Laufw*:][*Kennung*]

Typ: extern; bei Netzwerkbetrieb nicht erlaubt.

Bedeutung: Erzeugt, ändert oder löscht eine Diskkennung.

L*:*BPfad Laufwerk und Suchpfad für die LABEL–Befehlsdatei.

***Laufw*:** Eine gültige Laufwerksbezeichnung nach MS–DOS–Konventionen für das Laufwerk, in dem die Disk eingelegt ist, deren Kennung geändert werden soll. Wird ***Laufw*:** weggelassen, so nimmt MS–DOS das aktuelle Laufwerk an.

Kennung Eine beliebige Zeichenkette von 11 Zeichen Länge, die aus allen Zeichen, die auch für Dateinamen erlaubt sind (s.a. Kapitel 4.1.5), bestehen darf. Folgende Zeichen sind **nicht** erlaubt:

* ? / \ | . , ; : + = < > [ ]

Wird ***Kennung*** nicht angegeben, so zeigt MS–DOS folgendes an:

```
Volume in drive X is xxxxxxxxxxx

Volume label (11 characters, ENTER for
none)?_
```

(Disk in Laufwerke *X* ist *xxxxxxxxxxx*

Disk–Kennung (11 Zeichen, Eingabetaste–>keine Kennung)?). Wenn Sie die Disk–Kennung ändern wollen, können Sie jetzt eine neue Kennung eingeben und dann die Eingabetaste drücken. Wenn Sie nur die Eingabetaste drücken, fragt MS–DOS

```
Delete current volume label (Y/N)?_
```

(Aktuelle Diskkennung löschen (J/N)?). Wenn Sie jetzt die Y–Taste (Yes = Ja) drücken, wird die aktuelle Diskkennung gelöscht, bei N (No = Nein) bleibt sie erhalten.

Beispiel:

Wenn Sie eine Diskette mit Korrespondenz mit der Kennung BRIEFE 1986 versehen wollen, und diese Diskette befindet sich im Laufwerk A, so müssen Sie, wenn C das aktuelle Laufwerk mit den Systemdateien ist, folgendes eingeben:

```
label a:briefe 1986
```

Anmerkungen:

- LABEL darf nicht bei durch die Befehle SUBST oder ASSIGN anders zugeordneten Laufwerken verwendet werden.
- Man kann sich die Diskkennung auch mit dem Befehl DIR (s. dort in diesem Kapitel) anzeigen lassen.

MKDIR

Formate: **MKDIR** [***Laufw***:]***Pfad***

oder

MD [***Laufw***:]***Pfad***

Typ: intern

Bedeutung: Erzeugt ein neues, leeres Verzeichnis auf der Disk im angegebenen Laufwerk.

Laufw: Eine gültige Laufwerksbezeichnung nach MS–DOS–Konventionen für die Disk, auf der das Verzeichnis erzeugt werden soll. Wird ***Laufw***: weggelassen, so nimmt MS–DOS das aktuelle Laufwerk an.

Pfad Der Pfad (s.a. Kapitel 4.2.1) zu dem Verzeichnis, in dem das angegebene Verzeichnis erzeugt werden soll. Wird ***Pfad*** weggelassen, so erzeugt MKDIR das angegebene Verzeichnis im aktuellen Verzeichnis.

Beispiel: Wenn im Basisverzeichnis der Diskette im Laufwerk A: das Verzeichnis BRIEFE erzeugt werden soll, so müssen Sie eingeben:

```
md a:\briefe
```

Soll im Unterverzeichnis \BRIEFE auf der Diskette in Laufwerk A ein weiteres Unterverzeichnis namens MAI 86 erzeugt werden, so muß eingegeben werden:

```
md a:\briefe\mai 86
```

Anmerkungen:

- Für Pfadnamen gelten die in Kapitel 4.2.1.1 angegebenen Syntax–Regeln.
- Es können beliebig viele Unterverzeichnisse auf einer Disk erzeugt werden. Die Zahl ist nur durch den Speicherplatz der jeweiligen Disk beschränkt.
- Dateien und/oder Verzeichnisse dürfen denselben Namen haben, wenn sie in verschiedenen Verzeichnissen stehen.

MODE

Formate: für Paralleldrucker:

[*L*:][*BPfad*]**MODE LPT***n*[:][*Zch*][,[*Zeilen*][,P]]

für Datenfernübertragungsschnittstellen:

[*L*:][*BPfad*]**MODE**
COM*m*[:]*Baud*[,*Par*[,*Datenbits*[,*Stopbits*[,P]]]]

für die Umleitung der Paralleldruckerausgabe auf die Datenfernübertragungsschnittstelle (serielle Schnittstelle):

[*L*:][*BPfad*]**MODE LPT***n*[:]=**COM***m*[:]

für die Bildschirm–Anzeige:

[*L*:][*BPfad*]**MODE** *Anz*

oder

[*L*:][*BPfad*]**MODE** [*Anz*],*Shift*[,T]

Typ: extern

Bedeutung: Setzt den Betriebsmodus für Paralleldrucker, Datenfernübertragungsschnittstellen oder Adapter für Farb–/Grafik–Bildschirme.

L*:*BPfad Laufwerk und Suchpfad für die MODE–Befehlsdatei.

Die weiteren Parameter–Beschreibungen werden jetzt für die einzelnen o.a. Formate des MODE–Befehls getrennt angegeben.

Für Paralleldrucker:

n Ein ganzahliger Wert von 1, 2 oder 3, der die Drukker–Nummer angibt.

Zch Ein ganzzahliger Wert von 80 oder 132, der die Zeichenzahl pro Druckzeile angibt.

Zeilen Ein ganzzahliger Wert von 6 oder 8, der die Zeilenzahl pro Zoll angibt.

P Wird dieser Schalter gesetzt, so wiederholt MODE kontinuierlich die Ausgabe an den Drucker, wenn eine Gerätezeitüberschreitung vorgekommen ist. Mit diesem Schalter verbleibt ein Teil des MODE–Programms resident im Speicher. Eine durch **P** aufgebaute Zeit–Schleife kann durch Drücken der Tastenkombination Ctrl–Break abgebrochen werden.

Für Paralleldrucker sind folgende Werte eingestellt: LPT1, 80 Zeichen/Zeile, 6 Zeilen/Zoll.

Für Datenfernübertragungsschnittstellen:

m Ein ganzahliger Wert von 1 oder 2, der die Datenfernübertragungsschnittstellen–Nummer angibt.

Baud Ein ganzahliger Wert aus der Folge 110, 150, 300, 600, 1200, 2400, 4800, 9600, der die Übertragungsgeschwindigkeit in Baud angibt. Es müssen bei diesem Parameter mindestens die ersten beiden Ziffern des jeweiligen Wertes angegeben werden.

Par Einer der drei Buchstaben N, O, E für die Kennzeichnung der Paritätsprüfung.
N keine Paritätsprüfung (none)
O ungerade Prität (odd)
E gerade Parität (even)

Datenbits Ein ganzahliger Wert von 7 oder 8, der die Länge eines Datenwortes in Bits angibt.

Stopbits Ein ganzahliger Wert von 1 oder 2, der die Anzahl von Stopbits in einem Datenwort angibt.

P Wird dieser Schalter gesetzt, so wiederholt MODE kontinuierlich die Ausgabe an den Drucker, wenn eine Gerätezeitüberschreitung vorgekommen ist. Mit diesem Schalter verbleibt ein Teil des MODE–Programms resident im Speicher. Eine durch **P** aufgebaute Zeit–Schleife kann durch Drücken der Tastenkombination Ctrl–Break abgebrochen werden.

Für die Datenfernübertragungsschnittstelle sind folgende Werte voreingestellt: COM1, gerade Parität, 7 Datenbits, 2 Stopbits bei 110 Baud und 1 Stopbit bei allen anderen Baud–Raten.

Für die Umleitung der Paralleldruckerausgabe auf die Datenfernübertragungsschnittstelle:

n Ein ganzahliger Wert von 1, 2 oder 3, der die Drukker–Nummer angibt.

m Ein ganzahliger Wert von 1 oder 2, der die Datenfernübertragungsschnittstellen–Nummer angibt.

Wird die Paralleldruckerausgabe umgeleitet, verbleibt ein Teil des MODE–Programms resident im Hauptspeicher. Vor der Umleitung muß jedoch der Datenfernübertragungsmodus mit dem MODE–Befehl gesetzt werden.

Für die Bildschirm–Anzeige:

Anz Eine der folgenden Konstanten, mit der die Anzeigecharakteristiken des Bildschirms festgelegt werden: 40, 80, BW40, BW80, CO40, CO80, MONO. Es bedeuten:

40	40 Zeichen pro Bildschirmzeile.
80	80 Zeichen pro Bildschirmzeile.
BW40	aktiviert den Farb–/Grafik–Adapter für schwarz–weiße Anzeige bei 40 Zeichen pro Bildschirmzeile.

CO40 aktiviert den Farb–/Grafik–Adapter für farbige Anzeige bei 40 Zeichen pro Bildschirmzeile.

BW80 aktiviert den Farb–/Grafik–Adapter für schwarz–weiße Anzeige bei 80 Zeichen pro Bildschirmzeile.

CO80 aktiviert den Farb–/Grafik–Adapter für farbige Anzeige bei 80 Zeichen pro Bildschirmzeile.

MONO aktiviert den monochromen Bildschirmadapter mit konstanter Zeilenlänge von 80 Zeichen.

Shift Eine Buchstabenkonstante (L oder R), die angibt, nach welcher Richtung die Bildschirmanzeige zur exakten Justierung verschoben werden soll. L bedeutet links, R rechts. Wird dieser Parameter angegeben, so verbleibt ein Teil des MODE–Programmes resident im Hauptspeicher.

T Wird dieser Schalter angegeben, so wird in der ersten Bildschirmzeile eine Zahlenreihe für die Justierung der Bildschirmanzeige dargestellt. Außerdem fragt MODE mit

```
Do you see the rightmost 9 (Y/N)?_
```

(Sehen Sie die äußerst rechte 9 (J/N)?), ob die Anzeige sauber zentriert ist. Wird hier mit N (No = Nein) geantwortet, so wird die Anzeige in die mit ***Shift*** festgelegte Richtung verschoben und die o.a. Frage wiederholt. Bei Y (Yes = Ja) wird der MODE–Befehl beendet.

Beispiele:

Wenn Sie Ihre Druckerausgabe an einen Drucker mit serieller Schnittstelle senden wollen, benötigen Sie den MODE–Befehl zweimal. Einmal, um den Datenfernübertragungsmodus zu setzten und ein zweites Mal, um die Paralleldruckerausgabe, die standardmäßig gesetzt ist, auf die Datenfernübertragungsschnittstelle umzuleiten. Wenn Ihr Drucker z.B. mit einer Übertragungsrate von 4800 Baud bei gerader Parität arbeitet und an der Schnittstelle COM1 angeschlossen ist, müssen Sie die

beiden folgenden Befehle in dieser Reihenfolge eingeben:

```
mode com1:48,e,,,p
mode lpt1:=com1:
```

Wenn Sie dann jedoch wieder den Paralleldrucker LPT1 als Ausgabegerät setzen wollen, brauchen Sie nur noch anzugeben:

```
mode lpt1:
```

Angenommen, Sie wollen jetzt auf einem Paralleldrucker drucken, der an der zweiten Parallelschnittstelle LPT2 angeschlossen ist. Der Ausdruck soll mit 80 Zeichen/Zeile bei 8 Zeilen/Zoll erfolgen. Dann müssen Sie eingeben:

```
mode lpt2: 80,8
```

oder

```
mode lpt2:,8
```

Wenn Sie die Druckausgabe solange wiederholen wollen, bis der Drucker bereit ist, müssen Sie folgendes eingeben:

```
mode lpt2: 80,8,p
```

Bei Störungen können Sie die so aktivierte endlose Zeitschleife durch Drücken der Ctrl–Break–Tastenkombination abbrechen. Wenn Sie den o.a. Befehl ohne den P–Schalter eingeben, wird die Wiederholung der Ausgabe bei nicht bereitem Drucker inaktiviert.

Anmerkungen:

- Wenn Sie Dateien jedesmal ausdrucken müssen, wenn Sie MS–DOS starten, können Sie den MODE-Befehl auch in die AUTOEXEC.BAT–Datei aufnehmen. Näheres dazu finden Sie im Kapitel 5.6.

MORE

Format: [*L*:][*BPfad*]**MORE**

Typ: extern

Bedeutung: MORE ist ein Filterbefehl (s.a. Kapitel 5.5.2), der das "Durchblättern" einer Textausgabe auf dem Bildschirm in 23–Zeilen–Blöcken erlaubt. Er liest dabei die Eingabe von der Standard–Eingabeeinheit. Nach der Ausgabe von 23 Zeilen auf dem Bildschirm zeigt MORE am unteren Bildschirmrand die Meldung

```
--More--
```

an, und pausiert mit der Ausgabe der nächsten 23 Bildschirmzeilen, bis eine beliebige Taste gedrückt wird.

L*:*BPfad Laufwerk und Suchpfad für die MORE–Befehlsdatei.

Beispiel: Angenommen, Sie wollen die sehr lange Textdatei REPORT.TXT auf der Disk in Laufwerk A in Ruhe nach einer bestimmten Passage durchsuchen. Dann können Sie eingeben:

```
type a:report.txt | more
```

Dasselbe Ergebnis erhalten Sie, wenn Sie die Standardeingabe von der Datei REPORT.TXT auf den MORE–Filter umleiten:

```
more < report.txt
```

PATH

Format: **PATH** [[***Laufw***:]***Pfad***[[;[***Laufw***:]***P fad***]. . .]]

oder

PATH ;

Typ: intern

Bedeutung: Legt fest, welche Verzeichnisse nach dem aktuellen Verzeichnis außerdem nach externen Befehlen (.COM–, .BAT– oder .EXE–Dateien) durchsucht werden sollen. Voreingestellt ist kein Suchpfad. Der PATH–Befehl ohne Parameter zeigt den aktuellen Suchpfad an.

Laufw: Eine gültige Laufwerksbezeichnung nach MS–DOS–Konventionen für das/die zu durchsuchende(n) Laufwerk(e).

Pfad Der/die Pfad(e) (s.a. Kapitel 4.2.1) für das/die zu durchsuchende(n) Verzeichnis(se). Es können mehrere Suchpfade angegeben werden, die durch Semikolon (;) voneinander getrennt werden müssen.

Wird als Parameter nur ein Semikolon angegeben (PATH ;), so wird nur das aktuelle Verzeichnis nach externen Befehlen durchsucht (NUL–Pfad).

Beispiel: Wenn Sie MS–DOS anweisen wollen, externe Befehle in den drei Verzeichnissen \BENUTZER\RUDOLF, B:\BENUTZER\MARIA und \BIN zu suchen, können Sie folgenden PATH–Befehl eingeben:

```
path \benutzer\rudolf;b:\benutzer
\maria;\bin
```

Anmerkungen:

- Fehlerhaft angegebene Pfade (z.B. falsche Laufwerksangaben oder Trennzeichen) werden erst entdeckt, wenn MS–DOS den Pfad durchsucht.

- Nicht mehr existierende, aber im PATH–Befehl angegebene Pfade ignoriert MS–DOS bei der Suche.

- In durch PATH angegebenen Verzeichnissen findet MS–DOS nur .COM–, .EXE– oder .BAT–Programmdateien. Programmdateien mit anderen oder keinen Namenserweiterungen werden nicht gefunden.

PRINT

Format: [*L*:][*BPfad*]**PRINT** [[*Laufw*:][*Pfad*] [*Datei*[.*Erw*]. . .] [**/D:***Gerät*] [**/B:***Größe*][**/Q:***Wert*] [**/T**][**/C**][**/P**]

Typ: extern

Bedeutung: Überträgt die angegebene Liste von Dateien in eine Warteschlange (Druckpuffer) und druckt dann die Dateien im Hintergrund auf einem angeschlossenen Drucker aus. Während des Druckens können andere MS-DOS-Befehle oder Anwenderprogramme aufgerufen und abgearbeitet werden. Wird PRINT ohne Parameter aufgerufen, so wird nur der Inhalt der Warteschlange angezeigt, ohne den ggf. laufenden Druckvorgang zu beeinflussen.

L:BPfad Laufwerk und Suchpfad für die PRINT–Befehlsdatei.

Laufw:PfadDatei.Erw. . . Die vollständige(n) Angabe(n) für alle zu druckenden Dateien.

/D:*Gerät* Für ***Gerät*** kann der MS–DOS–Name für das Ausgabegerät angegeben werden. Erlaubt sind LPT1, LPT2, LPT3, PRN, COM1, COM2 und AUX. Wird der /D–Schalter nicht angegeben, so fordert PRINT mit der Anzeige

```
Name of list device [PRN]: _
```

(Name des List–Gerätes [PRN]) die Eingabe eines Gerätenamens. Wird nur die Eingabetaste gedrückt, so nimmt PRINT das Gerät PRN (Paralleldrucker LPT1) an.

Die nächsten beiden Schalter dürfen nur beim ersten Aufruf des PRINT–Befehls nach dem Starten von MS–DOS angegeben werden:

/B:*Größe* Mit diesem Schalter kann die voreingestellte Puffergröße von 512 Bytes vergrößert werden. Insbesondere beim Ausdruck vieler oder großer Dateien beschleunigt dies die Ausgabe.

Q/:*Wert* Wenn mehr oder weniger als 10 Dateien in die Warteschlange aufgenommen werden sollen, kann mit diesem Schalter die Anzahl verändert werden. Minimal darf /Q:4 und maximal /Q:32 angegeben werden.

/T Wird dieser Schalter angegeben, so werden alle noch in der Warteschlange befindlichen Dateien, also die, die noch nicht gedruckt wurden, aus dieser entfernt.

/C Mit diesem Schalter wird der Löschmodus gesetzt. Es werden dann der vorhergehende und alle folgenden in der Befehlszeile angegebenen Dateinamen aus der Warteschlange entfernt, bis der /P–Schalter (s.u.) in der Befehlszeile gefunden wird oder die Eingabetaste gedrückt wird.

/P Mit diesem Schalter wird der Druckmodus gesetzt. Es werden dann der vorhergehende und alle folgenden in der Befehlszeile angegebenen Dateinamen in die Warteschlange übernommen, bis der /C–Schalter (s.o.) in der Befehlszeile gefunden wird oder die Eingabetaste gedrückt wird.

Beispiele: Der folgende Befehl wird zum erstenmal nach dem Starten von MS–DOS eingegeben:

```
print a:report.txt
```

Es erscheint die Anzeige

```
Name of list device [PRN]: _
```

(Name des List–Gerätes [PRN]). Durch Drücken der Eingabetaste wird der angegebene Dateinamen in die Warteschlange genommen und die Ausgabe erfolgt auf dem Paralleldrucker.

Der Befehl

```
print /t
```

löscht alle Einträge in der Warteschlange.

Wenn Sie z.B. alle Dateieinträge mit dem Namen REPORT und beliebiger Namenserweiterung im aktuellen Verzeichnis der Disk im aktuellen Laufwerk aus der Warteschlange löschen wollen und außerdem die Datei BRIEF.TXT aus dem Verzeichnis \KORRESP auf der Disk in Laufwerk A in die Warteschlange eintragen wollen, müssen Sie eingeben:

```
print report.* /c
print a:\korresp\brief.txt /p
```

Anmerkungen:

- Jeder Warteschlangeneintrag darf maximal 64 Zeichen lang sein. Wenn Sie Dateien aus einer sehr tief geschachtelten Verzeichnisstruktur ausdrucken wollen, müssen Sie deshalb vorher ggf. das Verzeichnis mit dem CHDIR–Befehl (s. dort in diesem Kapitel) wechseln.

- Bei Dateinamen und Namenserweiterungen sind die Jokerzeichen * und ? für globale Dateinamen erlaubt. Wenn mehrere Dateinamen in der Befehlszeile angegeben werden, werden die zugehörigen Schalter /C und /P hinter dem jeweiligen Dateinamen angegeben.

- Sind Dateinamen in der Warteschlange eingetragen, kann das aktuelle Verzeichnis mit dem CHDIR–Befehl jederzeit gewechselt werden, ohne den Druckvorgang zu beeinflussen.

- Wenn bei der Befehlseingabe keiner der Parameter /T, /C oder /P gesetzt wird, so werden alle Dateinamen in die Warteschlange gesetzt (/P ist also Voreinstellung).

- Es sollte vor der Benutzung des PRINT–Befehl sichergestellt werden, daß das Ausgabegerät auch

angeschlossen und eingeschaltet ist. Andernfalls kann es zum Systemabsturz kommen und MS–DOS muß neu gestartet werden.

- Die Dateien werden in derselben Reihenfolge in die Warteschlange eingetragen und dann auch ausgedruckt, in der sie in der Befehlszeile angegeben wurden. Am Ende jeder Datei wird ein Seitenvorschub auf dem Drucker ausgeführt. TAB–Zeichen führen zur Tabulierung (Auffüllen mit Leerzeichen) bis zur nächsten 8–Spalten–Position.

- Tritt beim Lesen einer Datei von einer Disk ein Disk-Fehler auf, verfährt PRINT folgendermaßen:

  1. Der Eintrag für die momentan gedruckte Datei wird aus der Warteschlange gestrichen.

  2. Auf dem Drucker wird eine entsprechende Fehlermeldung zusammen mit dem betroffenen Dateinamen ausgedruckt.

  3. Der Drucker führt einen Seitenvorschub aus und meldet den Fehler außerdem mit einem Piep-Ton.

  4. Die nächste Datei aus der Warteschlange wird versucht auszudrucken.

- Sollen gerade ausgedruckte Dateien mit einem PRINT–Befehl zusammen mit dem Schaltern /T oder /C gelöscht werden, verfährt PRINT folgendermaßen:

  1. Der Drucker erzeugt einen Piep–Ton.

  2. Wurde der Schalter /C verwendet, so wird die Meldung

     ***Datei* canceled by operator**

     (*Datei* vom Bediener gelöscht) ausgedruckt, wobei für *Datei* der bezogene Dateiname gesetzt

wird. Wurde der Schalter /T verwendet, so wird die Meldung

```
All files canceled by operator
```

(Alle Dateien vom Bediener gelöscht) ausgedruckt.

3. Auf dem Drucker wird ein Seitenvorschub ausgeführt.

4. Falls nicht alle Einträge in der Warteschlange gelöscht wurden, wird der Ausdruck mit der ersten noch in der Warteschlange stehenden Datei fortgesetzt.

- Während des Ausdruckens darf die Diskette mit den zu druckenden Dateien nicht aus dem Laufwerk genommen werden. Auch darf keine der auszudruckenden Dateien während des Ausdrucks geändert oder gelöscht werden. Die anderweitige Benutzung des Druckers z.B. zum Bildschirmausdruck mit der PrtSc–Taste muß während des Ausdruckes unterbleiben.

PROMPT

Format: **PROMPT** [***Bereitschaftsanzeige***][$[***Zeichen***]]

Typ: intern

Bedeutung: Ändert die Standard–MS–DOS–Bereitschaftsanzeige. Wird PROMPT ohne Parameter benutzt, so wird wieder die Standard–Bereitschaftsanzeige ***akt. Laufwerk***> gewählt.

Bereitschaftsanzeige Ein beliebiger Text, der jedesmal vor dem Cursor angezeigt wird, wenn MS–DOS bereit ist, Befehle zu empfangen.

$*Zeichen* Mit Hilfe des Dollarzeichens in Verbindung mit bestimmten anderen Zeichen können bestimmte Informationen mit in die Bereitschaftsanzeige aufgenommen werden. Wird nach dem $–Symbol kein weiteres Zeichen angegeben, so besteht die Bereitschaftsanzeige nur aus dem blinkenden Cursor. Folgende Zeichen können nach dem $–Symbol angegeben werden:

Zeichen	Bereitschaftsanzeige
$	das $ –Zeichen.
t	die aktuelle Uhrzeit.
d	das aktuelle Datum.
p	das aktuelle Verzeichnis der Disk im aktuellen Laufwerk.
v	die MS–DOS–Versionsnr.
n	das aktuelle Laufwerk
g	das > –Zeichen.
l	das < –Zeichen.
b	das | –Zeichen.
q	das = –Zeichen.
h	Rückschritt und Löschen des vorhergehenden Zeichens.
e	ASCII–Code 1BH (ESC–Zeichen).
_	ASCII–Code–Folge 0DH 0AH (carriage return/line feed).

Alle anderen Zeichen, die nach dem $–Symbol angegeben werden, werden ignoriert. Die Angaben ***Bereitschaftsanzeige*** und ***$Zeichen*** können auch kombiniert werden.

Beispiel:

Um vor der Standard–Bereitschaftsanzeige die aktuelle Zeit anzuzeigen, können Sie folgenden PROMPT–Befehl verwenden:

```
prompt $t$h$h$h $n$g
```

Um eine zweizeilige Bereitschaftsanzeige der Form:

```
Uhrzeit = akt. Zeit
Datum = akt. Datum
```

zu erzeugen, müssen Sie eingeben:

```
prompt Uhrzeit = $t$_Datum = $d
```

Wenn der ANSI–Bildschirm–Gerätetreiber installiert ist (s. Kapitel 2.5.1), können Sie die Bereitschaftsanzeige z.B. in Form des invers dargestellten aktuellen Laufwerks definieren. Alle Eingaben werden dagegen in normaler Darstellung abgebildet:

```
prompt $e[7m$n:$e[m
```

RECOVER

Format: [*L*:][*BPfad*]**RECOVER** [*Laufw*:][*Pfad*] *Datei*[*.Erw*]

oder

[*L*:][*BPfad*]**RECOVER** [*Laufw*:]

Typ: extern; bei Netzwerkbetrieb nicht erlaubt.

Bedeutung: Stellt Dateien mit defekten Sektoren oder komplette Disks mit defekten Verzeichnissen wieder her.

L*:*BPfad Laufwerk und Suchpfad für die RECOVER-Befehlsdatei.

Laufw*:*PfadDatei.Erw Die vollständige Angabe für die wiederherzustellende Datei. Wird ***Laufw*:** nicht angegeben, so nimmt MS–DOS das aktuelle Laufwerk an. Wird ***Pfad*** nicht angegeben, so nimmt MS–DOS das aktuelle Verzeichnis an. Die Daten in dem oder den defekten Sektoren werden allerdings nicht wiederhergestellt.

Beim zweiten Format wird die gesamte Disk im angegebenen oder im aktuellen Laufwerk wiederhergestellt.

Beispiele: Wenn Sie festgestellt haben, daß z.B. in der Datei REPORT.TXT im Basisverzeichnis der Diskette im Laufwerk A fehlerhafte Sektoren vorhanden sind, können Sie diese Datei mit folgendem Befehl wiederherstellen:

```
recover a:\report.txt
```

Im allgemeinen ist die wiederhergestellte Datei größer als die ursprüngliche.

Wenn z.B. das Basisverzeichnis der Disk im aktuellen Laufwerk defekt ist, brauchen Sie nur einzugeben:

```
recover
```

RECOVER durchsucht dann die Dateizuordnungstabelle der Disk im aktuellen Laufwerk nach Folgen von Zuordnungseinheiten. Für jede gefundene Folge erzeugt RECOVER dann einen neuen Eintrag mit dem Namen FILE*nnnn*.REC im Basisverzeichnis der Disk, wobei *nnnn* die laufende Nr. der wiederhergestellten Datei ist und mit 0001 beginnt.

Anmerkungen:

- Das Format RECOVER [*Laufw*:] sollte nur bei zerstörten Diskverzeichnissen angewendet werden, da RECOVER nicht erkennen kann, ob die Daten im Verzeichnis selbst gültig sind. RECOVER geht deshalb immer von einem vollständig zerstörten Verzeichnis aus, obwohl vielleicht nur ein Sektor unleserlich ist. Deshalb werden grundsätzlich alle Dateien einer solchen Disk wiederhergestellt.

- Wenn eine Textdatei wiederhergestellt wurde, ist meistens eine neue Edierung des Bereiches der Daten erforderlich, der im zerstörten Sektor lag.

- Da RECOVER in dem Format mit Dateinamen immer nur eine Datei wiederherstellt, ist die Verwendung der Jokerzeichen * und ? für globale Dateinamen hier sinnlos.

- RECOVER markiert bei der Wiederherstellung einer Datei oder Disk die defekten Sektoren, damit diese von MS–DOS beim weiteren Gebrauch der Disk nicht mehr zur Speicherung verwendet werden.

RENAME

Format: **REN[AME]** [***Laufw***:][***Pfad***]***Datei***[***.Erw***] ***Datei***[***.Erw***]

Typ: intern

Bedeutung: Gibt der ersten angegebenen Datei den zweiten angegebenen Namen. Für den Befehlsnamen genügt die Eingabe von REN.

Laufw:PfadDatei.Erw Die vollständige Angabe für die umzubenennende Datei.

Datei.Erw Der neue Name für die umzubenennende Datei.

Beispiele: Mit dem folgenden Befehl wird die Namenserweiterung aller Dateien im aktuellen Verzeichnis der Disk im aktuellen Laufwerk, deren Namen mit .TXT enden. in .DOC geändert:

```
ren *.txt *.doc
```

Wenn die Datei KAP10 im aktuellen Verzeichnis der Disk im Laufwerk B in TEIL10 umbenannt werden soll, müssen Sie eingeben:

```
ren b:kap10 teil10
```

Anmerkungen:

- Der RENAME–Befehl benennt **alle** Dateien um, deren Namen mit dem ersten im Befehl angegebenen Namen übereinstimmen. Da aber logischerweise eine Umbenennung über verschiedene Laufwerke hinweg nicht möglich ist, erzeugt RENAME eine Fehlermeldung, wenn beim zweiten Dateinamen eine Laufwerksbezeichnung angegeben wird.

- Sowohl bei der ersten Dateiangabe als auch beim zweiten Dateinamen dürfen die Jokerzeichen * und ? für globale Dateinamen verwendet werden. Werden Sie beim zweiten Dateinamen verwendet, wird die

Position der zugehörigen Zeichen jedoch nicht verändert.

- Beim zweiten Dateinamen kann **kein** Suchpfad angegeben werden. Die umbenannte Datei verbleibt also in ihrem ursprünglichen Verzeichnis.

REPLACE

Format: [*L*:][*BPfad*]**REPLACE** [*Laufw*:][*Pfad*] *Datei*[.*Erw*] [*Laufw*:]*Pfad* [/A][/D][/P][/R][/S][/W]

Typ: extern

Bedeutung: Ersetzt eine Datei durch eine aktuellere Version.

L*:*BPfad Laufwerk und Suchpfad für die REPLACE–Befehlsdatei.

Laufw*:*PfadDatei.Erw Die vollständige Angabe für die aktuelle Datei. Die Jokerzeichen * und ? für globale Dateinamen sind erlaubt. Auf diese Weise können auch mehrere aktuelle Dateien angegeben werden.

Pfad Der Suchpfad für das Ziel–Verzeichnis, in dem eine ältere Version einer Datei zu ersetzen ist.

/A Wird dieser Schalter benutzt, so werden die angegebenen Dateien zum Zielverzeichnis hinzugefügt, statt die älteren Versionen dort zu ersetzen. Dieser Schalter darf **nicht** zusammen mit dem /D– oder dem /S–Schalter (s.u.) benutzt werden.

/D Wird dieser Schalter benutzt, so werden die Dateien im Zielverzeichnis nur dann ersetzt, wenn die angegebenen Dateien zeitlich aktueller sind. Dieser Schalter ist inkompatibel zum /A–Schalter (s.o.).

/P Wird dieser Schalter benutzt, so wird vor dem Ersetzen oder Hinzufügen von Dateien die Anfrage

```
Replace Dateiname? (Y/N)_
```

(*Dateiname* ersetzen? (J/N) angezeigt. Für *Dateiname* wird die bei der Befehlseingabe angegebene Dateibezeichnung gesetzt. Es wird nur ersetzt oder hinzugefügt, wenn mit Y (Yes = Ja) geantwortet wurde. Andernfalls wird REPLACE beendet.

/R Wird dieser Schalter benutzt, so werden auch schreibgeschützte (Dateiattribut R (read only)) Dateien ersetzt. Fehlt der /R–Schalter, so führt der Versuch, eine schreibgeschützte Datei zu ersetzen, zu einer Fehlermeldung und zur Beendigung des REPLACE–Befehls.

/S Wird dieser Schalter benutzt, so werden alle Dateien in allen Unterverzeichnissen des angegebenen Zielverzeichnisses ersetzt, deren Name mit dem der angegegebenen aktuellen Datei übereinstimmen. Im Pfad der aktuellen Datei durchsucht REPLACE dagegen keine Unterverzeichnisse. Dieser Schalter ist inkompatibel zum /A–Schalter (s.o.).

/W Wird dieser Schalter benutzt, so wartet REPLACE mit dem Ersetzen oder Hinzufügen, bis der Benutzer eine beliebige Taste gedrückt hat, andernfalls beginnt REPLACE sofort mit der Arbeit.

Beispiele:

Angenommen, auf Ihrer Festplatte befinden sich in verschiedenen Verzeichnissen mehrere alte Versionen einer Kundentelefondatei namens TELEFON.KDN, die Sie alle durch die neueste Version, die sich im Basisverzeichnis der Disk in Laufwerk A befindet, ersetzen wollen. Dann müssen Sie eingeben:

```
replace a:\telefon.kdn c:\ /s
```

Angenommen, Sie wollen zu einem Unterverzeichnis namens \MSTOOLS auf der Festplatte, in dem Druckertreiber für ein Textprogramm abgelegt sind, weitere Druckertreiber hinzufügen, die alle die Namenserweiterung .DBS haben. Dann können Sie folgenden Befehl eingeben:

```
replace a:*.dbs c:\mstools /a
```

Anmerkungen:

- Der REPLACE–Befehl hat zwei Funktionen:

  1. Er ersetzt alle Dateien im Zielverzeichnis, die denselben Namen haben, wie die angegebene aktuelle Datei. Dies ist die voreingestellte Funktion.

2. Wahlweise fügt REPLACE auch die angebene aktuelle Datei zum Zielverzeichnis hinzu (/A–Schalter (s.o.)).

- Während REPLACE arbeitet, werden die Namen der betroffenen Dateien auf dem Bildschirm angezeigt. Am Ende werden je nach Erfolg folgende Meldungen angezeigt:

  ***nnn* file(s) added/replaced**

  (*nnn* Datei(en) hinzugefügt/ersetzt) oder

  No files added/replaced

  (Keine Datei hinzugefügt/ersetzt).

- Verborgene Dateien oder Systemdateien können mit dem REPLACE–Befehl **nicht** ersetzt werden.

- Der REPLACE–Befehl erzeugt bei seiner Beendigung die folgenden Fehlergrad–Codes, die mit dem Stapelverarbeitungsbefehl IF ERRORLEVEL (s. dort in Kapitel 6.3.2) verarbeitet werden können, wenn REPLACE in einer Stapeldatei verwendet wird:

0	Normale Beendigung; kein Fehler.
1	Fehler in der Befehlszeile.
2	Datei nicht gefunden.
3	Pfad nicht gefunden.
5	Dateizugriff verboten.
8	Speicherplatz nicht ausreichend.
15	Ungültige Laufwerksangabe.
andere	Standard–MS–DOS–Fehler.

RESTORE

Format: [*L*:][*BPfad*]**RESTORE** *Laufw*:[*Laufw*:][*Pfad*]*Datei*[*.Erw*] [**/S**][**/P**][**/A:***Datum*][**/B:***Datum*][**/E:***Zeit*][**/L:***Zeit*][**/M**][**/N**]

Typ: extern

Bedeutung: Speichert die angegebenen Sicherungsdateien, die mit dem BACKUP–Befehl (s. dort in diesem Kapitel) erstellt wurden, auf die Disk im angegebenen Laufwerk zurück.

L:BPfad Laufwerk und Suchpfad für die RESTORE–Befehlsdatei.

Laufw: Die erste Laufwerksangabe bezieht sich auf das Laufwerk, in dem die Disk mit den BACKUP–Dateien eingelegt ist (Quell–Disk).

Laufw:PfadDatei.Erw Die vollständige Dateiangabe für alle Dateien, die zurückgespeichert werden sollen. Dabei bezieht sich die Angabe ***Laufw:Pfad*** auf die Zieldisk und das Zielverzeichnis und ***Datei.Erw*** auf die zurückzuspeichernden Dateien.

/S Wird dieser Schalter benutzt, so werden auch alle angegebenen Dateien aus allen Unterverzeichnissen in allen Ebenen unterhalb des angegebenen Verzeichnisses zurückgespeichert.

/P Wird dieser Schalter benutzt, so fragt RESTORE bei jeder schreibgeschützten, verborgenen oder seit der letzten Rückspeicherung veränderten Datei, deren Name mit dem angegebenen übereinstimmt, ob diese zurückgespeichert werden soll oder nicht.

/A:*Datum* Es werden nur Dateien zurückgespeichert, die am angegebenen Datum oder später verändert wurden.

/B:*Datum* Es werden nur Dateien zurückgespeichert, die am angegebenen Datum oder früher verändert wurden.

/E:*Zeit* Es werden nur Dateien zurückgespeichert, die zur angegebenen Zeit oder früher verändert wurden.

/L:*Zeit* Es werden nur Dateien zurückgespeichert, die zur angegebenen Zeit oder später verändert wurden.

/M Es werden nur Dateien zurückgespeichert, die seit der letzten Sicherung mit BACKUP verändert wurden.

/N Es werden nur Dateien zurückgespeichert, die nicht mehr auf der Zieldisk existieren.

Beispiele:

Wenn alle Dateien in allen Unterverzeichnissen auf allen Sicherungsdisketten, die nacheinander in Laufwerk A eingelegt werden sollen, auf die Festplatte in Laufwerk C zurückgespeichert werden sollen, müssen Sie eingeben:

```
restore a: c:\*.* /s
```

Der folgende Befehl speichert alle Dateien mit der Namenserweiterung .TXT von der Sicherungsdiskette in Laufwerk A in das aktuelle Verzeichnis der Festplatte in Laufwerk C zurück:

```
restore a: c:*.dat
```

Wenn Sie alle Dateien vom Sicherungslaufwerk A auf das aktuelle Laufwerk zurückspeichern wollen und dabei feststellen wollen, welche Dateien schreibgeschützt oder verborgen sind oder seit der letzten Sicherung verändert wurden, können Sie eingeben:

```
restore a:\/p
```

Anmerkungen:

- RESTORE kann nur Dateien zurückspeichern, die vorher mit dem BACKUP–Befehl gesichert wurden.

- Wird für das Ziel kein Pfad angegeben, so bestimmt das aktuelle Verzeichnis die Rückspeicherung. Wenn ein Pfad angegeben wird, muß auch ein Dateiname angegeben werden.

- Damit auch mehrere Dateien mit übereinstimmenden Namensteilen zurückgespeichert werden können, dürfen bei der Dateiangabe auch die Jokerzeichen * und ? für globale Dateinamen verwendet werden.

- RESTORE fordert den Benutzer auf, die Sicherungsdisketten in der richtigen Reihenfolge einzulegen. Diese Aufforderung wird wiederholt, wenn die angegebene(n) Datei(en) sich nicht auf der eingelegten Diskette befinden.

- Wenn gemeinsamer Dateizugriff (Netzwerkbetrieb) aktiviert ist, können nur die Dateien zurückgespeichert werden, auf die der Benutzer Zugriff hat. Andernfalls wird eine Fehlermeldung angezeigt.

- Bei globalen Dateinamen wird der Benutzer aufgefordert, die nächste Sicherungsdiskette einzulegen, wenn RESTORE von der vorhergehenden alle Dateien, auf die der globale Dateiname zutrifft, zurückgespeichert hat.

- Dateien, die mit einer älteren MS–DOS–Version gesichert wurden, sollten nur unter Benutzung des /P–Schalters zurückgespeichert werden. Damit können dann die beiden verborgenen Systemdateien MSDOS.SYS und IO.SYS erkannt und übergangen werden.

- Der RESTORE–Befehl erzeugt bei seiner Beendigung die folgenden Fehlergrad–Codes, die mit dem Stapelverarbeitungsbefehl IF ERRORLEVEL (s. dort in Kapitel 6.3.2) verarbeitet werden können, wenn REPLACE in einer Stapeldatei verwendet wird:

  0 Normale Beendigung; kein Fehler.
  1 Es wurden keine zurückzuspeichernden Dateien gefunden.

2 Es wurden einige Dateien wegen Problemen beim gemeinsamen Dateizugriff nicht zurückgespeichert.
3 Abbruch durch den Benutzer mit Ctrl–Break.
4 Abbruch durch Fehler.

RMDIR

Formate: **RMDIR** [***Laufw***:]***Pfad***

oder

RD [***Laufw***:]***Pfad***

Typ: intern

Bedeutung: Löscht ein bis auf die beiden .– und . .–Einträge leeres Verzeichnis auf der Disk im angegebenen Laufwerk.

Laufw: Eine gültige Laufwerksbezeichnung nach MS–DOS–Konventionen für die Disk, auf der das Verzeichnis gelöscht werden soll. Wird ***Laufw***: weggelassen, so nimmt MS–DOS das aktuelle Laufwerk an.

Pfad Der Pfad (s.a. Kapitel 4.2.1) zu dem Verzeichnis, das gelöscht werden soll.

Beispiel: Wenn im Basisverzeichnis der Diskette im Laufwerk A: das Verzeichnis BRIEFE gelöscht werden soll, so müssen Sie eingeben:

```
rd a:\briefe
```

Das Verzeichnis BRIEFE muß bis auf die beiden .– und . .–Einträge leer sein.

Anmerkungen:

- Mit dem RMDIR–Befehl können weder das aktuelle noch das Basisverzeichnis einer Disk gelöscht werden.

SET

Format: **SET** [***Name***=[***Zeichenkette***]]

Typ: intern

Bedeutung: Fügt eine Zeichenkette in die Umfeld–Zeichentabelle des Befehlsprozessors für die spätere Verwendung bei Befehlen oder Anwenderprogrammen ein.

Name Eine Bezeichnung für die nachfolgende Zeichenkette. Kleinbuchstaben werden hier in Großbuchstaben umgewandelt. Ist ***Name*** bereits in der Umfeld–Zeichentabelle vertreten, so wird der Rest durch die neu angegebene ***Zeichenkette*** ersetzt.

Zeichenkette Eine beliebige Zeichenfolge, die so, wie sie angegeben wird, hinter ***Name*** in die Umfeldzeichentabelle eingetragen wird. Wird nur ***Name***= ohne ***Zeichenkette*** angegeben, so wird die Kombination ***Name***=***Zeichenkette*** aus der Umfeldzeichentabelle des Befehlsprozessors gelöscht.

Wird der SET–Befehl ohne Parameter benutzt, so zeigt MS–DOS den Inhalt der gesamten Umfeldzeichentabelle an.

Beispiel: Sie können in einer Stapeldatei die austauschbaren Parameter mit Hilfe des SET–Befehls durch einen Namen anstatt durch eine Nummer definieren. Wenn z.B. Ihre Stapeldatei den Befehl

```
type %file%
```

enthält, können Sie mit dem SET–Befehl den Namen setzen, den MS–DOS bei Abarbeiten der Stapeldatei für den austauschbaren Parameter %FILE% setzen soll, also z.B.

```
set file=report.mai
```

Anmerkungen:

- Die gesamte Umfeldzeichentabelle wird allen MS–DOS–Befehlen und allen Anwenderprogrammen zur Verfügung gestellt. Aus diesem Grunde sollte der SET–Befehl nur benutzt werden, wenn ein Anwenderprogramm die Umfeldzeichentabelle benutzen soll.

- Zusätzlich zu allen mit SET angegebenen Zeichenfolgen fügt MS–DOS automatisch die bei einem benutzten PROMPT– oder PATH–Befehl angegebenen Parameter zusammen mit dem Namen PROMPT= bzw. PATH= in die Umfeldzeichentabelle ein.

- MS–DOS setzt beim Start als erste Zeichenfolge unter dem Namen COMSPEC= den Pfad ein, auf dem der Befehlsprozessor COMMAND.COM gefunden werden kann, wenn er nachgeladen werden soll.

- Hat MS–DOS noch **kein** speicherresidentes Programm geladen, wie z.B. Teile des MODE– oder PRINT–Befehls oder ein Tastaturprogramm oder die Befehle GRAPHICS bzw. GRAFTABL, so wird der Bereich der Umfeldzeichentabelle von MS–DOS zur Aufnahme weiterer Zeichenketten erweitert. Ist dagegen bereits ein speicherresidentes Programm geladen, so wird der Umfang der Umfeldzeichentabelle auf 127 Bytes begrenzt. MS–DOS gibt eine Fehlermeldung aus, wenn versucht wird, mit einem SET-Befehl Zeichenfolgen in die Umfeldzeichentabelle zu bringen, die deren Umfang auf über 127 Bytes ausdehnen würde, **nachdem** ein speicherresidentes Programm geladen wurde.

- Die Umfeldzeichentabelle ist immer dann nützlich, wenn der Anwender für sein Programm Parameter setzen will, die für MS–DOS keine Bedeutung haben, vom Anwenderprogramm aber interpretiert werden können.

SHARE

Format: [*L*:][*BPfad*]**SHARE** [**/F:***Bereich*][**/L:***Sperren*]

Typ: extern

Bedeutung: Lädt die Unterstützung für gemeinsamen Dateizugriff und gegenseitige Zugriffsverriegelung bei Netzwerkbetrieb.

L:BPfad Laufwerk und Suchpfad für die SHARE–Befehlsdatei.

/F:*Bereich* Zusammen mit dem /F–Schalter wird die Größe des Bereiches in Bytes festgelegt, in dem MS–DOS interne Informationen bei aktiviertem gemeinsamen Dateizugriff ablegt. Für jede bei gemeinsamen Dateizugriff geöffnete Datei wird Platz für den vollen Dateinamen und zusätzlich 11 Bytes benötigt. Voreingestellt ist ein Wert von 2048 Bytes.

/L:*Sperren* Zusammen mit dem /L–Schalter kann die maximale Anzahl erlaubter Sperren für Dateizugriffe angegeben werden. Voreingestellt ist hier ein Wert von 20.

Anmerkungen:

- Nachdem der gemeinsame Dateizugriff mit dem SHARE–Befehl installiert ist, prüft MS–DOS bei jeder Schreib–/Lese–Operation, ob es sich um einen erlaubten Zugriff handelt.

- Die Installation des gemeinsamen Dateizugriffs ist nur einmal möglich und bleibt bis zum Abschalten des Computers resident.

SORT

Format: [*L*:][*BPfad*]**SORT** [/**R**][/+*n*]

Typ: extern

Bedeutung: SORT ist ein Filterbefehl, der Daten vom Standardeingabegerät zeilenweise liest, sie auf die angegebene Art und Weise alphanumerisch sortiert und an das Standardausgabegerät ausgibt.

L*:*BPfad Laufwerk und Suchpfad für die SORT–Befehlsdatei.

/R Wird dieser Schalter angegeben, so erfolgt die Sortierung alphabetisch absteigend, also von Z bis A. Voreingestellt ist alphabetisch aufsteigende Sortierung.

/+*n* Wird dieser Schalter angegeben, so wird jede Zeile erst ab Spalte ***n*** sortiert. ***n*** ist ein ganzzahliger Wert und mit 1 voreingestellt.

Beispiele: Wenn Sie z.B. die Datei TELEFON.TXT im aktuellen Verzeichnis der Disk im aktuellen Laufwerk alphabetisch absteigend sortieren und das Ergebnis in der Datei TELEFON.SRT im selben Verzeichnis ablegen wollen, können Sie eingeben:

```
sort /r <telefon.txt >telefon.srt
```

Wenn Sie das aktuelle Verzeichnis der Disk im aktuellen Laufwerk nach aufsteigender Dateigröße sortieren wollen, und das Ergebnis bildschirmweise ansehen wollen, können Sie zwei Filterbefehle kombinieren:

```
dir | sort /+14 | more
```

Anmerkungen:

- Sortiert wird grundsätzlich nach dem ASCII–Code der Daten. Der Code für Kleinbuchstaben wird für die Sortierung (**nicht** bei den Ausgabedaten) in den für Großbuchstaben gewandelt. Die Umlaute, Sonder-

buchstaben und Sonderzeichen der europäischen Sprachen, deren Codes zwischen 128 dez und 175 dez sowie bei 225 dez liegen, werden entsprechend der u.a. Tabelle für die Sortierung durch Großbuchstaben oder Standardsonderzeichen ersetzt (Code–Angaben dezimal):

Code	Ersatzcode	Code	Ersatzcode
128	67	153	79
129	85	154	85
130	69	155	36
131	65	156	36
132	65	157	36
133	65	158	36
134	65	159	36
135	67	160	65
136	69	161	73
137	69	162	79
138	69	163	85
139	73	164	78
140	73	165	78
141	73	166	166
142	65	167	167
143	65	168	63
144	69	169	169
145	65	170	170
146	65	171	171
147	79	172	172
148	79	173	33
149	79	174	34
150	85	175	34
151	85	225	83
152	89		

- Wenn Standard–Ein– und –Ausgabe in Diskdateien umgeleitet werden (s. erstes Beispiel oben), so müssen sich die beiden Dateinamen unterscheiden.

- Die maximale Größe der zu sortierenden Datei darf 63 KBytes nicht übersteigen.

SUBST

Formate: [*L*:][*BPfad*]**SUBST** ***Laufw*: *Laufw*:*Pfad***

oder

[*L*:][*BPfad*]**SUBST** ***Laufw*:/D**

oder

[*L*:][*BPfad*]**SUBST**

Typ: extern; bei Netzwerkbetrieb nicht erlaubt

Bedeutung: Ersetzt einen Laufwerksbuchstaben oder eine Kombination aus Laufwerksbuchstaben und Zugriffspfad durch einen anderen Laufwerksbuchstaben.

L*:*BPfad Laufwerk und Suchpfad für die SUBST–Befehlsdatei.

***Laufw*:** Ein beliebiger Buchstabe, der zukünftig das danach angegebene Laufwerk und ggf. den Pfad ersetzt.

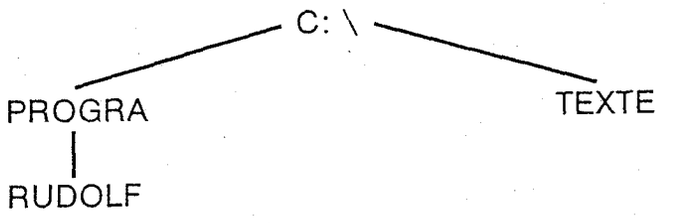
Laufw:Pfad Die Laufwerks– oder komplette Pfadbezeichnung, für die die erste Laufwerksbezeichnung (s.o.) zukünftig gelten soll.

***Laufw:*/D** Die Laufwerkssubstitution wird wieder aufgehoben. Für ***Laufw*:** muß die Bezeichnung eingegeben werden, die die ursprüngliche Laufwerks– oder Pfad–Bezeichnung ersetzt hat.

Wird der SUBST–Befehl ohne Parameter angegeben, so werden alle gegenwärtig gesetzten Laufwerks–Substitutionen angezeigt.

Beispiele:

Bei den im folgenden angegebenen Beispielen sei A das aktuelle Laufwerk und das Verzeichnis RUDOLF das aktuelle Verzeichnis der Festplatte in Laufwerk C, die folgende Verzeichnis–Struktur habe:



Außerdem sei in der CONFIG.SYS–Datei der LASTDRIVE-Befehl (s.a. Kapitel 2.3) mit LASTDRIVE = R gesetzt und die externen MS–DOS–Befehlsdateien befinden sich im Basisverzeichnis der Festplatte.

Der Zugriffspfad zum aktuellen Verzeichnis RUDOLF auf der Festplatte kann jetzt folgendermaßen durch die Laufwerksbezeichnung Q ersetzt werden:

```
c:\subst q: c:\progra\rudolf
```

Zum Anzeigen des aktuellen Verzeichnisses von der Festplatte brauchen Sie dann nur noch anzugeben:

```
dir q:
```

Die Eingabe

```
c:\subst
```

ergibt für das o.a. Beispiel dann die Anzeige

```
q: => c:\progra\rudolf
```

Soll diese Substitution wieder aufgehoben werden, muß eingegeben werden:

```
c:\subst q: /d
```

Anmerkungen:

- Die Laufwerkssubstitution ist überall da sinnvoll, wo Anwenderprogramme nicht die Angabe eines Zugriffspfades erlauben, so z.B. Anwenderprogramme, die noch unter MS–DOS–Versionen älter als 2.0 erstellt wurden.

- MS–DOS zeigt in folgenden Fällen die Meldung

  invalid parameter

  (ungültiger Parameter) an:

  – Falsche Laufwerksangabe.
  – Das aktuelle Laufwerk soll substituiert werden.
  – Die Substitutionslaufwerksbezeichnung ist ein Buchstabe, der weiter hinten im Alphabet steht, als der mit dem LASTDRIVE–Befehl in der CONFIG.SYS–Datei (s.a. Kapitel 2.3) gesetzte (voreingestellt ist E).
  – Der Schalter /D wird ohne Laufwerksangabe benutzt.
  – Beide Laufwerksangaben sind gleich.

- Wird ein Pfad angegeben, so sollte dieser grundsätzlich beim Basisverzeichnis beginnen.

- Bei substituierten Laufwerken oder Pfaden sollten die Befehle CHDIR, MKDIR, RMDIR und PATH für die bezogenen Laufwerke mit Vorsicht verwendet werden.

- Die Befehle ASSIGN, BACKUP, DISKCOMP, DISKCOPY,FDISK, FORMAT, JOIN, LABEL und RESTORE dürfen bei aktiven Substitutionen **nicht** verwendet werden.

- Ein Substitutions–Laufwerk kann als virtuelles Laufwerk angesehen werden, also als Laufwerk, das physikalisch nicht existiert.

SYS

Format: [*L*:][*BPfad*]**SYS** ***Laufw*:**

Typ: extern; bei Netzwerkbetrieb nicht erlaubt

Bedeutung: Überträgt die beiden verborgenen MS–DOS–Systemdateien MSDOS.SYS und IO.SYS in das Basisverzeichnis der Disk im angegebenen Laufwerk.

L*:*BPfad Laufwerk und Suchpfad für die SYS–Befehlsdatei.

***Laufw*:** Eine gültige Laufwerksangabe nach MS-DOS-Konventionen für die Disk, auf die die MS–DOS–Systemdateien übertragen werden sollen.

Anmerkungen

- Bei SYS muß ein Ziellaufwerk angegeben werden.
- Die Zieldiskette muß entweder vollständig leer sein oder aber nur bereits dasselbe System enthalten, das übertragen werden soll.
- Befinden sich auf der Zieldiskette bereits die beiden MS–DOS–Systemdateien MSDOS.SYS und IO.SYS, so müssen diese genau denselben Speicherplatz belegen, wie das übertragene System. Das bedeutet also, daß Sie die Systemdateien von MS–DOS 3.20 nicht auf eine Disk mit den Systemdateien von DOS 1.1 oder 2.1 übertragen können. Sie müssen vielmehr vorher diese Disk mit dem Formatbefehl von MS–DOS 3.20 formatieren.
- Die Systemdateien werden in der Reihenfolge

  1. IO.SYS
  2. MSDOS.SYS

  übertragen.
- SYS überträgt **nicht** den Befehlsprozessor COMMAND.COM.

TIME

Format: **TIME** [***hh.mm.***[***.ss***[,nn]]]

Typ: intern

Bedeutung: Setzt die System–Uhr auf die angegebene Zeit oder zeigt die aktuelle Systemzeit an.

hh Eine Zahl zwischen 0 und 23 für die aktuelle Stunde.

mm Eine Zahl zwischen 0 und 59 für die Minuten.

ss Eine Zahl zwischen 0 und 59 für die Sekunden.

nn Eine Zahl zwischen 0 und 99 für hundertstel Sekunden.

Anmerkungen:

- Wird der TIME–Befehl ohne Parameter angegeben, so zeigt MS–DOS die Meldung

  Current time is *hh.mm.ss,nn*
  Enter new time:_

  (Aktuelle Zeit ist *hh.mm.ss,nn*
  Neue Zeit eingeben:_) an. Jetzt kann eine neue Uhrzeit im angezeigten Format eingegeben oder die angezeigte Zeit kann durch Drücken der Eingabetaste bestätigt werden. Solange der Rechner eingeschaltet bleibt, werden Datum und Uhrzeit von der Systemuhr weitergeführt.

- Das Format der Zeit wird vom COUNTRY–Befehl in der CONFIG.SYS–Datei bestimmt (s.a. Kapitel 2.3). Die Stunden–, Minuten– und Sekundenangaben können voneinander durch Punkte (.) oder Doppelpunkte (:) getrennt werden. Werden hundertstel Sekunden angegeben, so müssen diese je nach COUNTRY–Code in der CONFIG.SYS–Datei mit einem Komma (,) oder Punkt (.) von den Sekunden getrennt werden.

- Verfügen Sie über einen Commodore PC 40, so braucht bei diesem Rechner die Uhrzeit nur ein einziges Mal mit Hilfe des Programms SETUP (s. Anhang A im *MS–DOS–Bedienungshandbuch*) gesetzt zu werden, da Datum und Uhrzeit mit einer batteriegepufferten Uhr auch dann weitergeführt werden, wenn der Rechner ausgeschaltet ist. Wenn hier jedoch die Uhrzeit mit dem TIME–Befehl geändert wird, so gilt diese Änderung nur solange, wie der Rechner nicht ausgeschaltet wird. Die batteriegepufferte Systemuhr kann also nur mit dem Programm SETUP gestellt werden.

TREE

Format: [*L*:][*BPfad*]**TREE** [*Laufw*:][**/F**]

Typ: extern

Bedeutung: Zeigt für das angegebene Laufwerk für jedes Verzeichnis einschließlich aller Unterverzeichnisse den Zugriffspfad an und listet wahlweise alle Dateien in allen Verzeichnissen.

L*:*BPfad Laufwerk und Suchpfad für die TREE–Befehlsdatei.

***Laufw*:** Eine gültige Laufwerksbezeichnung nach MS–DOS–Konventionen für die Disk, deren Zugriffspfade angezeigt werden sollen. Wird ***Laufw*:** nicht angegeben, so nimmt MS–DOS das aktuelle Laufwerk an.

/F Wird dieser Schalter angegeben, so zeigt TREE zu jedem Zugriffpfad auch alle Dateinamen des jeweiligen Verzeichnisses an.

Beispiel: Wenn Sie eine komplette Liste der Verzeichnis– und Dateistruktur der Diskette in Laufwerk A ausdrucken wollen, brauchen Sie nur einzugeben:

```
tree a: /f > prn
```

TYPE

Format: **TYPE** [***Laufw***:][***Pfad***]***Datei***[***.Erw***]

Typ: intern

Bedeutung: Listet den Inhalt der angegebenen (Text–)Datei auf dem Bildschirm.

Laufw: Eine gültige Laufwerksbezeichnung nach MS–DOS–Konventionen für die Disk, auf der die anzuzeigende Datei gespeichert ist. Wird ***Laufw***: weggelassen, so nimmt MS–DOS das aktuelle Laufwerk an.

Pfad Der Pfad (s.a. Kapitel 4.2.1), auf dem die bezogene Datei zu finden ist. Wird ***Pfad*** weggelassen, so nimmt MS–DOS das aktuelle Verzeichnis an.

Datei[***.Erw***] Der vollständige Name der Datei, deren Inhalt angezeigt werden soll. Hat die Datei eine Namenserweiterung, so muß diese auch angegeben werden. Die Jokerzeichen * und ? für globale Dateinamen sind hier **nicht** erlaubt.

Beispiele: Wenn die Assembler–Quellsprachedatei PROG.ASM aus dem Unterverzeichnis PROGRA im aktuellen Laufwerk auf dem Drucker ausgedruckt werden soll, müssen Sie eingeben:

```
type \progra\prog.asm > prn
```

Anmerkungen:

- TYPE zeigt nur Dateien mit ASCII–Text an. Wenn mit TYPE eine Binärdatei (z.B. eine Programmdatei) angezeigt wird, erhält man nicht–interpretierbare Zeichen auf dem Bildschirm.

- TYPE expandiert Tabulatorzeichen (09H) durch 8 Leerstellen.

- Parallel zur Bildschirmausgabe kann die Datei auch ausgedruckt werden, wenn vor der Eingabe des

TYPE–Befehls die Tastenkombination Ctrl–Umschalttaste–PrtSc gedrückt wird. Nach dem Ausdruck wird durch nochmaliges Drücken die Parallelausgabe auf den Drucker wieder abgeschaltet. In diesem Fall braucht also die Ausgabe **nicht** wie bei dem o.a. Beispiel mit dem Zusatz > PRN extra auf den Drucker umgeleitet zu werden.

- Ebenso wie auf den Drucker kann die Ausgabe von TYPE auch in eine Datei umgeleitet werden (s.a. Kapitel 5.5.1).

VER

Format: **VER**

Typ: intern

Bedeutung: Zeigt die aktuelle MS–DOS–Versionsnummer auf dem Bildschirm an.

Anmerkungen:

- Wenn Sie

  ```
  ver
  ```

  eingeben, erhalten Sie die Anzeige

  ```
  MS-DOS Version 3.20
  ```

  3 ist die Haupt–Versionsnummer, 20 die aktuelle Ausgabe der Version 3.

VERIFY

Format: **VERIFY [ON | OFF]**

Typ: intern

Bedeutung: Schaltet die Lesen–nach–Schreiben–Prüfung für das Speichern von Daten auf Disk ein (ON) oder aus (OFF). Wird die Prüfung eingeschaltet, so prüft MS–DOS nach jedem physikalischen Schreibvorgang auf der Disk, ob die geschriebenen Daten auch richtig gespeichert wurden. Voreinstellung ist OFF. Bei ON verlangsamt sich der Ablauf von solchen Programmen, die viele Schreibzugriffe auf Disk ausführen. Wird VERIFY ohne Parameter angegeben, so zeigt MS–DOS den gegenwärtigen Zustand des VERIFY–Schalters.

Beispiele: Die Eingabe von

```
verify on
```

schaltet die Schreibprüfung ein.

Nach dem Start von MS–DOS führt die Eingabe von

```
verify
```

zur Anzeige

```
VERIFY is off
```

Anmerkungen:

- Wurde VERFY ON gegeben, so bleibt dieser Zustand solange aktuell, wie kein VERIFY OFF eingegeben wurde oder kein MS–DOS–Systemaufruf SET VERIFY benutzt wurde.

VOL

Format: **VOL** [***Laufw***:]

Typ: intern

Bedeutung: Zeigt die Diskkennung (falls vorhanden) der Disk im angegebenen Laufwerk auf dem Bildschirm an.

Laufw: Eine gültige Laufwerksangabe nach MS–DOS-Konventionen für die Disk, deren Kennung angezeigt werden soll. Wird ***Laufw***: nicht angegeben, so nimmt MS–DOS das aktuelle Laufwerk an.

Beispiel: Angenommen, Sie haben bei der Formatierung Ihrer Festplatte im Laufwerk C mit dem FORMAT–Befehl zusammen mit dem /V–Schalter die Diskkennung TEXTE vergeben, so führt die Eingabe von

```
vol c:
```

zur Anzeige

```
vol in drive C is TEXTE
```

Anmerkungen:

- Hat die Disk im angegebenen Laufwerk keine Kennung, so zeigt MS–DOS an:

  ```
  Volume in drive x has no label.
  ```

- Mit dem LABEL–Befehl (s. dort in diesem Kapitel) können Sie jederzeit die Diskkennung jeder Disk neu eintragen, ändern oder löschen.

XCOPY

Format: [*L*:][*BPfad*]**XCOPY** [*Laufw*:][*Pfad*]*Datei*[.*Erw*]
[*Laufw*:][*Pfad*][*Datei*[.*Erw*]]
[/**A**][/**D**:*Datum*]
[/**E**][/**M**][/**P**][/**S**][/**W**][/**V**]

Typ: extern

Bedeutung: Kopiert die angegebene Quelle (Dateien oder komplette Verzeichnisse) auf das angegebene Ziel (Dateien oder Verzeichnisse).

L*:*BPfad Laufwerk und Suchpfad für die XCOPY–Befehlsdatei.

Laufw*:*PfadDatei.Erw Die erste Angabe spezifiziert die Datei(en) oder das Verzeichnis, das kopiert werden soll. Es muß mindestens ein Parameter angegeben werden. Wird nur ***Laufw*:** angegeben, so werden alle Dateien im aktuellen Verzeichnis der Disk im Quellaufwerk kopiert. Wird nur ***Pfad*** angegeben, so werden alle Dateien des angegebenen Verzeichnisses der Disk im aktuellen Laufwerk kopiert. Wird nur ***Datei.Erw*** angegeben, so wird nur diese eine Datei im aktuellen Verzeichnis der Disk im aktuellen Laufwerk kopiert. Hier sind die Jokerzeichen ? und * für globale Dateinamen erlaubt.

Laufw*:*PfadDatei.Erw Die zweite Angabe spezifiziert das Kopierziel. Werden alle Ziel–Parameter weggelassen, so wird die Quelle in das aktuelle Verzeichnis der Disk im aktuellen Laufwerk kopiert. Wird nur ***Datei.Erw*** beim Ziel weggelassen, so fragt XCOPY mit der Meldung

```
Does Pfad specify a file name
or directory name on the target
(F = file, D = directory)? _
```

(Bezeichnet *Pfad* eine Datei oder ein Verzeichnis auf dem Ziel (F = Datei; D = Verzeichnis)?), ob die Pfadangabe sich

auf eine Datei oder ein Verzeichnis bezieht. Wird mit F geantwortet, so wird die Quelle in die angegebene Zieldatei kopiert. Wird mit D geantwortet, so sucht XCOPY zunächst das Zielverzeichnis. Ist es nicht vorhanden, wird es erzeugt. Anschließend wird die Quelle in das Ziel kopiert.

/A Wird dieser Schalter gesetzt, so werden nur die Dateien der Quelle kopiert, bei denen das Archiv–Attributbit gesetzt ist (s.a. ATTRIBUT–Befehl in diesem Kapitel).

/D:*Datum* Wird dieser Schalter gesetzt, so werden nur die Dateien der Quelle kopiert, die am angegebenen Datum oder später verändert wurden. Das Datum muß entsprechend dem mit dem COUNTRY–Befehl in der CONFIG.SYS–Datei gesetzten Code (s.a. Kapitel 2.3) angegeben werden.

/E Wird dieser Schalter gesetzt, so werden auch alle Unterverzeichnisse der Quelle kopiert, selbst wenn sie leer sind. Dieser Schalter muß zusammen mit dem /S–Schalter (s.u.) benutzt werden.

/M Dieser Schalter entspricht insoweit dem /A–Schalter, als auch hier nur Quell-Dateien mit gesetztem Archiv-Attributbit kopiert werden. Gleichzeitig wird aber bei der kopierten Datei das Archiv–Bit gelöscht.

/P Wird dieser Schalter gesetzt, so erwartet XCOPY vor dem Kopieren jeder Quell–Datei eine Bestätigung (Y/N–Antwort).

/S Wird dieser Schalter gesetzt, so werden auch alle Unterverzeichnisse der Quelle kopiert, es sei denn, sie sind leer. Wird dieser Schalter nicht gesetzt, so arbeitet XCOPY grundsätzlich nur mit einem einzelnen Verzeichnis.

/V Wird dieser Schalter gesetzt, so werden alle kopierten Daten mit den Quelldaten verglichen, um die Integrität der kopierten Daten zu gewährleisten.

/W Wird dieser Schalter gesetzt, so meldet XCOPY vor Beginn des Kopiervorganges

```
Press any key when ready to start
copying files
```

(Beliebige Taste zum Starten drücken) und beginnt erst mit dem Kopieren, wenn eine Taste gedrückt wird. Drücken von Ctrl–Break bricht XCOPY ab.

Beispiele:

Da der DISKCOPY–Befehl Disketten spurweise kopiert, müssen Quell– und Ziel–Disks dasselbe Format haben. Wenn Sie dagegen eine Disk mit Dateien in Unterverzeichnissen haben, und Sie wollen diese Disk auf eine Disk mit anderem Format kopieren, so können Sie dies nur mit dem XCOPY–Befehl. So kopiert das folgende Beispiel alle Dateien und Unterverzeichnisse (einschließlich aller leeren Unterverzeichnisse) der Quelldisk in Laufwerk A auf die Zieldisk in Laufwerk B:

```
xcopy a: b: /s /e
```

Wenn Sie die Anfrage von XCOPY, ob es sich bei dem angegebenen Zielpfad um ein Verzeichnis oder eine Datei handelt (s.o. bei der Beschreibung der Zielparameter), unterdrücken wollen, können Sie folgendermaßen vorgehen:

Erstellen Sie sich mit dem Befehl

```
copy /b xcopy.exe mcopy.exe
```

eine modifizierte Befehlsdatei MCOPY.EXE, die automatisch bestimmt, ob der angegebene Zielpfad eine Datei oder ein Verzeichnis ist. MCOPY arbeitet nach folgenden Regeln:

- Wenn die Quelle ein Verzeichnis ist, ist auch das Ziel ein Verzeichnis.
- Bezeichnen die Quellparameter mehrere Dateien, so ist das Ziel ein Verzeichnis.

- Endet die Namensangabe beim Ziel mit einem linken Schrägstrich (\), so ist das Ziel ein Verzeichnis. Das folgende Beispiel erzeugt im Basisverzeichnis der Disk in Laufwerk A das Unterverzeichnis REPORTS und kopiert die Datei BERICHT.MAI aus dem aktuellen Verzeichnis der Disk im aktuellen Laufwerk in dieses Verzeichnis:

```
mcopy bericht.mai a:\reports\
```

Anmerkungen:

- Der XCOPY–Befehl erzeugt bei der Beendigung die folgenden Fehlergrad–Codes, die mit dem Stapelverarbeitungsbefehl IF ERRORLEVEL (s. dort in Kapitel 6.3.2) verarbeitet werden können:

  0 normale Beendigung; kein Fehler.
  1 keine zu kopierenden Dateien gefunden.
  2 Abbruch durch den Benutzer mit Ctrl–Break.
  4 Speichermangel, Syntaxfehler in der Befehlszeile, Datei oder Pfad nicht gefunden.
  5 Abbruch mit Systeminterrupt 24, der auf einen fatalen Schreib–/Lese–Fehler hinweist.

6.3.2 Stapelverabeitungsbefehle

Im Kapitel 5.6 werden die Möglichkeiten der Stapelverarbeitung von MS–DOS–Befehlen in Stapeldateien beschrieben. Mit Hilfe der im folgenden beschriebenen speziellen Stapelverarbeitungsbefehle lassen sich regelrechte Programme erzeugen, deren Befehle aus einer Kombination von MS–DOS–Befehlen und Stapelverarbeitungs–Befehlen bestehen und die in einer Stapeldatei abgelegt werden. Stapelverarbeitungsbefehle sind besondere interne MS–DOS–Befehle, die in Stapeldateien benutzt werden können. Folgende Stapelbefehle stehen zur Verfügung:

ECHO	schaltet die Anzeige der Befehle in der Stapeldatei ein oder aus oder zeigt Nachrichten an.
FOR. .IN. .DO	erlaubt Schleifenverarbeitung.
GOTO	erlaubt Verzweigungen in der Stapeldatei.
IF	erlaubt die bedingte Aufführung von Befehlen in der Stapeldatei.
PAUSE	hält die Abarbeitung einer Stapeldatei an.
REM	zeigt Kommentarzeilen aus einer Stapeldatei an.
SHIFT	erlaubt die Verwendung von mehr als 10 austauschbaren Parametern in einer Stapeldatei.

Im folgenden wird jeder dieser Befehle in dem schon bei den allgemeinen MS–DOS–Befehlen benutzten Format detailliert beschrieben.

ECHO

Format: **ECHO [ON | OFF | *Meldung*]**

Typ: intern

Bedeutung: Erlaubt (ECHO ON) oder verbietet (ECHO OFF) die Anzeige des gerade aus einer Stapeldatei ausgeführten MS–DOS–Befehls während der Abarbeitung der Stapeldatei.

Meldung Eine beliebige anwenderspezifische Meldung, die angezeigt werden soll. ***Meldung*** kann beliebige Zeichen enthalten. Es brauchen keine Einkleidungszeichen (z.B. ") angegeben zu werden. Ist ***Meldung*** länger als eine Bildschirmzeile, so muß vor jeder neuen Meldungszeile das Befehlswort ECHO wiederholt werden.

Wird ECHO ohne Parameter benutzt, so wird der gegenwärtige ECHO–Status (ON oder OFF) angezeigt.

Beispiel: Eine mehrzeilige Meldung in einer Stapeldatei könnte so aussehen:

```
echo off
echo
Dieses Stapelverarbeitungsprogramm
echo formatiert und überprüft
echo neue Disketten
```

Anmerkungen:

- Die Version ECHO *Meldung* des ECHO–Befehls ist nur in einer Stapeldatei sinnvoll, nachdem ECHO OFF gegeben wurde.

FOR. . .IN. . .DO

Formate: Bei Stapelverarbeitung:

FOR %%*Variable* IN (*Satz*) DO *Befehl*

Bei interaktiver Verarbeitung im MS–DOS–Bereitschaftsmodus:

FOR %*Variable* IN (*Satz*) DO *Befehl*

Typ: intern

Bedeutung: Erlaubt die wiederholte Ausführung von MS–DOS–Befehlen in Form einer einzeiligen Programmschleife. Explizit ausgedrückt:

Für jedes Element in *Satz* führe den *Befehl* aus.

%%Variable Für ***Variable*** kann der Benutzer jeden Buchstaben verwenden. Um Verwechslung mit den Austauschparametern %0 bis %9 zu vermeiden, dürfen keine Ziffern verwendet werden. Die ***%%Variable*** wird sequentiell auf jedes Element von ***Satz*** (s.u.) gesetzt. Danach wird der ***Befehl*** ausgeführt. Enthält ein Element von ***Satz*** die Jokerzeichen ? und/oder * für globale Dateinamen, so wird ***%%Variable*** auf jeden übereinstimmenden Dateinamen auf der Disk gesetzt. In diesem Fall steht nur ein einziges Element in diesem ***Satz***. ***Befehl*** ignoriert dann alle anderen Elemente.

Satz Ein Satz von Pfad– oder Dateinamen, getrennt durch Leerzeichen, die die zu verarbeitenden Dateien angeben, oder von beliebigen MS–DOS–Befehlen (auch Stapelbefehlen).

Befehl Ein beliebiger MS–DOS–Befehl, der entsprechend der Anzahl der in ***Satz*** angegebenen Elemente für jedes Element ausgeführt werden soll.

Beispiele:

Im folgenden Beispiel sollen Assembler–Quellsprachedateien assembliert werden. Dafür wird die Variable %F auf alle Dateien mit der Namenserweiterung ASM im aktuellen Verzeichnis der Disk im aktuellen Laufwerk angewendet:

```
for %%f in (*.asm) do masm %%f
```

Diese Stapelverarbeitungszeile führt einen Befehl des folgenden Formats aus: MASM *Datei*. Dabei werden z.B. folgende Dateien berücksichtigt: TRANSFER.ASM, INOUT.ASM, PRINT.ASM.

Beim nächsten Beispiel wird die Variable %F an die drei Dateien REPORT, NOTIZ und ADRESSEN gebunden, um diese drei Dateien zu löschen:

```
for %%f in (report notiz adresse) do del
%%f
```

Anmerkungen:

- In einer Befehlszeile darf nur ein FOR–Befehl benutzt werden, FOR–Befehle sind also nicht schachtelbar.

GOTO

Format: **GOTO *Marke***

Typ: intern

Bedeutung: Verzweigt innerhalb einer Stapelverarbeitungsdatei zu der Befehlszeile, die auf *Marke* folgt.

Marke Eine beliebige Buchstabenfolge, die auch Leerzeichen enthalten darf, und die das Sprungziel angibt. Das Sprungziel selbst muß in einer Zeile für sich stehen und mit einem Doppelpunkt (:) beginnen. Beim Sprungziel sind nur die ersten 8 Zeichen (ohne den Doppelpunkt) signifikant.

Beispiel: Im folgenden Beispiel wird zur Sprungmarke :ENDE verzweigt, wenn die Formatierung der Diskette im Laufwerk A fehlerlos verlaufen ist, andernfalls wird eine Meldung angezeigt:

```
:beginn
echo off
format a:/s
if errorlevel 0 goto end
echo Formatierungsfehler
:end
echo Ende der Stapelverarbeitung
```

Anmerkungen:

- Jede Befehlszeile einer Stapeldatei, die mit einem Doppelpunkt (:) beginnt, wird bei der Verarbeitung ignoriert, damit also auch nicht angezeigt. Deshalb können auf diese Weise auch Kommentare in eine Stapeldatei eingefügt werden.

- Wird in einer Stapeldatei zu einer nicht existierenden Sprungmarke verzweigt, so wird die Stapelverarbeitung beendet.

IF

Formate: **IF [NOT] ERRORLEVEL *Zahl Befehl***

oder

IF [NOT] ***Zeichenfolge1 == Zeichenfolge2 Befehl***

oder

IF [NOT] EXIST ***Datei*****[*****.Erw*****]** ***Befehl***

Typ: intern

Bedeutung: Erlaubt die bedingte Verarbeitung von Befehlen in einer Stapeldatei. ***Befehl*** wird nur ausgeführt, wenn die davor stehende Bedingung logisch wahr, d.h. erfüllt ist. Wird NOT angegeben, so wird ***Befehl*** ausgeführt, wenn die davor stehende Bedingung logisch falsch, d.h. **nicht** erfüllt ist. Ist die Bedingung nicht erfüllt, so wird der Befehl in der nächsten Zeile der Stapeldatei ausgeführt.

Zahl Die Bedingung ist erfüllt, wenn der zuletzt von COMMAND.COM ausgeführte Befehl mit einem Beendigungscode beendet wurde, der größer oder gleich ***Zahl*** ist.

Zeichenfolge1 == Zeichenfolge2 Die Bedingung ist erfüllt, wenn nach einer Parametersubstitution die beiden Zeichenfolgen identisch sind. Beide dürfen keine Trennzeichen wie Semikolon, Komma, Punkt, Gleichheits– oder Leerzeichen enthalten.

EXIST ***Datei.Erw*** Die Bedingung ist erfüllt, wenn die bei ***Datei.Erw*** angegebene Diskdatei existiert.

Beispiele:

Im folgenden Beispiel wird zur Sprungmarke :ENDE verzweigt, wenn die Formatierung der Diskette im Laufwerk A fehlerlos verlaufen ist, andernfalls wird eine Meldung angezeigt:

```
:beginn
echo off
format a:/s
if errorlevel 0 goto end
echo Formatierungsfehler
:end
echo Ende der Stapelverarbeitung
```

Das folgende Beispiel zeigt die Verwendung der Gleichheit zweier Zeichenfolgen. Wenn ein Parameter **Telefon** beim Aufruf der Stapeldatei verwendet wurde, zeigt die folgende Zeile eine entsprechende Meldung an:

```
if %1 == Telefon echo Telefon ist
vorhanden
```

Wenn bei einem Datei–bezogenen Befehl die Datei nicht existiert, wird eine Meldung angezeigt:

```
echo off
type %1
if not exist %1 echo Datei nicht
gefunden
```

Anmerkungen:

- Folgende MS–DOS–Befehle liefern einen Beendigungscode, der mit dem IF ERRORLEVEL–Stapelverarbeitungsbefehl ausgewertet werden kann:

  BACKUP
  DISKCOMP
  DISKCOPY
  REPLACE
  RESTORE
  XCOPY

PAUSE

Format: **PAUSE** [***Meldung***]

Typ: intern

Bedeutung: Unterbricht die Bearbeitung einer Stapeldatei, zeigt die ggf. angegebene Meldung an und wartet mit der Fortsetzung der Bearbeitung, bis eine beliebige Taste gedrückt wird.

Meldung Ein beliebiger Text bis zu einer Länge von 121 Zeichen, der angezeigt wird, sobald der PAUSE–Befehl wirksam wird.

Beispiel: Im folgenden Beispiel wird zum Einlegen einer Diskette in Laufwerk A aufgefordert. Die Formatierung beginnt, wenn eine beliebige Taste gedrückt wurde.

```
:beginn
pause Bitte Diskette in Laufwerk A
einlegen
format a:/s
if errorlevel 0 goto end
echo Formatierungsfehler
:end
echo Ende der Stapelverarbeitung
```

Man erhält die Anzeige:

```
Bitte Diskette in Laufwerk A einlegen
Strike a key when ready . . .
```

und nach dem Drücken einer Taste und der Formatierung

```
Ende der Stapelverarbeitung
```

Anmerkungen:

- Der PAUSE–Befehl zeigt neben der angegebenen Meldung noch den Text

  ```
  Strike a key when ready. . .
  ```

(Wenn bereit, Taste drücken. . .) an und wartet auf das Betätigen einer Taste. Wird jetzt die Tastenkombination Ctrl–Break oder Ctrl–C gedrückt, fragt MS–DOS mit der Meldung

```
Terminate batch job (Y/N)?
```

(Stapeljob beenden (J/N)?), ob die Bearbeitung der Stapeldatei beendet werden soll oder nicht. Wird mit Y (Yes = Ja) geantwortet, so kehrt MS–DOS auf die Befehlsebene zurück, andernfalls wird der nächste Befehl in der Stapeldatei bearbeitet. Diese Anfrage erscheint auch immer dann, wenn an beliebiger Stelle der Stapeldatei Ctrl–Break oder Ctrl–C gedrückt wird.

- Wurde vorher der Befehl ECHO OFF gegeben, so wird die ggf. hinter PAUSE angegebene Meldung **nicht** angezeigt, wohl aber die Aufforderung zum Betätigen einer Taste.

REM

Format: **REM** [***Kommentar***]

Typ: intern

Bedeutung: Erlaubt das Einfügen von kommentierendem Text in eine Stapeldatei.

Kommentar Ein beliebiger Text bis zu einer Länge von 123 Zeichen, der in derselben Zeile angezeigt wird, in der auch der REM–Befehl steht. Die einzigen Trennzeichen, die im Kommentartext erlaubt sind, sind Tabulator, Leerzeichen und Kommata (,). Wird ***Kommentar*** weggelassen, so wird eine Leerzeile in die Stapeldatei eingefügt, was zur besseren Lesbarkeit benutzt werden kann.

Beispiel: Im folgenden Beispiel wird der REM–Befehl sowohl zum Einfügen von Kommentar als auch von Leerzeichen benutzt:

```
rem Dieser Stapel formatiert und prüft
rem neue Disketten
rem Er wird mit CHECKNEU.BAT benannt
rem
pause Leerdiskette in Laufwerk A
einlegen
format a: /v
chkdsk a:
```

SHIFT

Format: **SHIFT**

Typ: intern

Bedeutung: Erlaubt die Verwendung von mehr als 10 austauschbaren Parametern (%0 bis %9) in einer Stapeldatei (s.a. Kapitel 5.6.3). Nach SHIFT werden alle Parameter in der Befehlszeile um 1 nach links verschoben, wobei Parameter %0 durch %1 ersetzt wird usw. Der nach %9 angegebene wird dann %9.

Beispiel: In der folgenden Stapeldatei KOPIER.BAT wird die Verwendung von SHIFT mit beliebig vielen Parametern gezeigt. Damit können alle im Aufruf angegebenen Dateien in ein angegebenes Verzeichnis kopiert werden:

```
remKOPIER.BAT kopiert alle angegebenen
rem Dateien in ein angegebenes
rem Verzeichnis.
rem der Aufruf ist
rem KOPIER Verzeichnis Datei,Datei....
rem
echo off
:eins
if "%1" == "" goto zwei
set zieldir = %1
shift
copy %1 %zieldir%
goto eins
:zwei
echo Fertig
```

Anmerkungen:

- Der SHIFT–Befehl darf auch verwendet werden, wenn weniger als 10 Parameter benutzt werden.
- **Achtung:** SHIFT kann nicht rückgängig gemacht werden. Wurde einmal SHIFT gegeben, so ist der vorher vorhandene Parameter %0 verloren.

7. MS–DOS–Edier– und Funktionstasten

Dieses Kapitel beschäftigt sich mit wichtigen Elementen der Tastatur, und wie diese von MS–DOS benutzt werden. Im einzelnen sind dies:

- Die MS–DOS–Edier– und Funktionstasten
- Der Befehlszeilenpuffer
- Die MS–DOS–Steuerzeichen

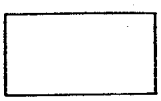
7.1 Spezielle MS–DOS–Edier– und –Bedienungstasten

Viele Betriebssysteme behandeln die Befehlseingabe anders als MS–DOS. Eine Besonderheit bei MS–DOS ist ein Satz spezieller Ediertasten. So ist es z.B. nicht erforderlich, einen sich häufig wiederholenden Befehl immer wieder neu einzugeben, denn MS–DOS überträgt den zuletzt eingegebenen Befehl in einen besonderen Befehlszeilenpuffer. Dieser Zeilenpuffer bietet zusammen mit den speziellen Ediertasten folgende Vorteile:

- Befehlswiederholung durch Drücken von nur 2 Tasten.
- Bei Eingabefehlern kann die letzte Befehlszeile problemlos mit Hilfe der Ediertasten bearbeitet werden.
- Die Edierung und Ausführung einer Befehlszeile, die der vorherigen ähnelt, ist mit einem Minimum an Aufwand möglich.

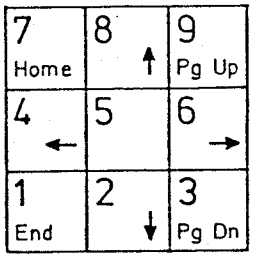
MS–DOS weist folgenden Tasten der Tastatur besondere Funktionen zur Bearbeitung der Befehlszeile zu:

Leertaste



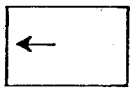
Die Leertaste setzt den Cursor (das ist der kleine blinkende Strich, der die aktuelle Schreibposition auf dem Bildschirm markiert) um eine Position nach rechts, nachdem an der aktuellen Schreibposition eine Leerstelle eingefügt wurde.

Cursor–Steuertasten



Die Cursor–Steuertasten (das sind die Tasten mit den Pfeilsymbolen auf der separaten Zehnertastatur rechts) dienen zum Positionieren des Cursors auf dem Bildschirm. Mit Ausnahme der ← – und → –Tasten haben diese Tasten im Befehlseingabemodus von MS–DOS keine Funktion.

Rücktaste



Die Rücktaste dient zum Korrigieren von Eingabefehlern, ehe die abschließende Eingabe–Taste gedrückt wird. Mit dieser Taste wird das links vom Cursor stehende Zeichen gelöscht, und der Cursor auf diese Position gestellt.

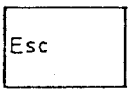
Einfüge–Taste

Schaltet den Einfügemodus ein oder aus. Bei eingeschaltetem Einfügemodus wird die Zeile auf dem Bildschirm um die eingegebenen Zeichen ergänzt, ohne daß der Zeichenzeiger im Eingabepuffer entsprechend weitergestellt wird.

Zeichen–Lösch–Taste

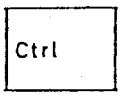
Löscht ein Zeichen im Eingabepuffer. Dieses Zeichen wird nicht angezeigt. Der Cursor in der Befehlszeile wird nicht bewegt.

Zeilen–Lösch–Taste



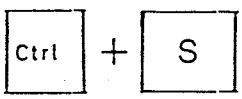
Löscht die aktuell angezeigte Befehlszeile vom Bildschirm. Der Inhalt des Eingabepuffers bleibt unverändert.

Ctrl–Taste



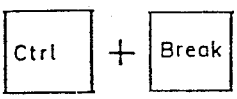
Die Ctrl–Taste hat eine sehr spezielle Funktion. Mit ihrer Hilfe können Sie komplexe Befehle an den Computer übermitteln, indem Sie nur zwei oder drei Tasten drükken. Wie die Umschalttaste auch muß die Ctrl–Taste niedergehalten werden, während Sie gleichzeitig eine oder zwei weitere Tasten drücken.

Bildschirm–rollen stoppen



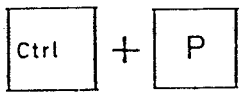
Wenn Sie gleichzeitig die Ctrl- und die S-Taste drücken, während Daten auf dem Bildschirm ausgegeben werden, wird die Anzeige angehalten und der Bildschirm nicht weiter nach oben gerollt. Dieselbe Funktion hat die Tastenkombination Ctrl–NumLock. Drücken einer beliebigen anderen Taste setzt die Bildschirmausgabe fort.

Befehlsausführung abbrechen



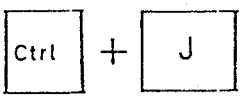
Die Tastenkombinationen Ctrl-C oder Ctrl-Break brechen die Ausführung eines vorher eingegebenen MS-DOS-Befehls ab.

Befehlsprotokollierung mit Drucker



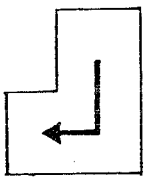
Alle eingegebenen Befehle werden parallel auf einem Drucker protokolliert, wenn die Tastenkombination Ctrl-P gedrückt wird. Nochmaliges Drücken schaltet die Protokollierung wieder aus.

Befehlszeilenverlängerung



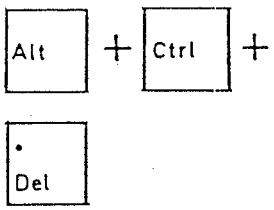
Wenn eine Befehlszeile nicht auf eine Bildschirmzeile paßt, kann am Ende des ersten Teiles die Tastenkombination Ctrl–J gedrückt werden. Damit wird auf dem Bildschirm ein Wagenrücklauf/Zeilenvorschub erzeugt, ohne daß der Befehlszeilenpuffer verändert wird. Die weiteren Eingaben werden dann dem ersten Teil der Befehlszeile hinzugefügt.

Eingabe–Taste



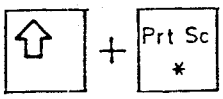
Die Eingabe– oder Return–Taste schließt die Eingabe eines Befehls ab und übermittelt die gesamte Eingabe zur Bearbeitung an MS–DOS. Zusätzlich wird die gesamte Eingabe in den Befehlszeilenpuffer kopiert.

MS–DOS neu starten



Wenn wegen irgendwelcher Störungen oder fehlerhafter Bedienungen das Rücksetzen des Rechnersystem erforderlich wird, so braucht der Computer nicht ab– und wieder angeschaltet zu werden. Vielmehr müssen die Tasten Ctrl–Alt–Del gleichzeitig niedergehalten werden.

Bildschirm ausdrucken



Wird diese Taste zusammen mit der Umschalttaste gedrückt, so wird der gesamte Bildschirm auf einem angeschlossenen Drucker ausgedruckt.

7.2 Funktionstasten

Neben den im letzten Abschnitt beschriebenen MS–DOS–Edier– und Bedienungstasten ordnet MS–DOS dem Funktionstastenblock links auf der Tastatur besondere Funktionen zu, mit denen die Befehlszeile, die nach dem Drücken der Eingabe–Taste von MS–DOS in einen Eingabepuffer kopiert wird, in diesem Eingabepuffer ediert oder auch wiederverwendet werden kann. Der Anwender spart dadurch Zeit, weil er bei wiederholter Benutzung einer Befehlszeile diese nicht noch einmal eingeben muß.

Generell gilt, daß bei normaler Befehlseingabe ohne Benutzung einer der unten beschriebenen Tasten der Zeichenzeiger im Eingabepuffer für jedes eingegebene Zeichen eine Position weitergestellt wird.

Im einzelnen gibt es folgende Funktionstasten zur Benutzung des MS–DOS–Befehlszeilenpuffers:

F1 oder 6 →	Zeigt ein Zeichen aus dem Eingabepuffer auf dem Bildschirm an.
F2	Kopiert alle Zeichen aus dem Eingabepuffer bis zu einem nach dem Drücken der F2–Taste eingegebenen Zeichen. Dieses Zeichen wird nicht mehr mit angezeigt.
F3	Kopiert alle noch im Eingabepuffer verbliebenen Zeichen in die Befehlszeile auf dem Bildschirm.
F4	Löscht alle Zeichen im Eingabepuffer bis zu einem nach Drücken der F4–Taste eingegebenen Zeichen. Das eingegebene Zeichen wird nicht mit gelöscht.
F5	Übernimmt die aktuelle Befehlszeile vom Bildschirm in den Eingabepuffer.
F6	Setzt ein Ctrl–Z–Zeichen (Dateiendekennung) ans Ende der Zeile im Eingabepuffer.
F7	Setzt ein Ctrl–@–Zeichen ans Ende der Zeile im Eingabepuffer.

7.3 Beispiele

Das folgende ausführliche Beispiel demonstriert die Benutzung der MS–DOS–Edier– und Funktionstasten.

Angenommen, Sie wollen sich den Verzeichniseintrag der Datei INVEST.MNT im aktuellen Verzeichnis der Disk im aktuellen Laufwerk ansehen. Dann können Sie eingeben:

```
dir invest.mnt
```

Nachdem Sie die Eingabetaste gedrückt haben, wird der Befehl unverzüglich ausgeführt und außerdem die Befehlszeile in den Zeilenpuffer kopiert. Wenn Sie dann diesen Befehl noch einmal ausführen lassen wollen, brauchen Sie nur die beiden Tasten **F3** und **Eingabe** drücken. Sobald Sie **F3** drücken, wird der Inhalt des Zeilenpuffers auf dem Bildschirm angezeigt, also

```
dir invest.mnt
```

Durch anschließendes Drücken der Eingabetaste wird dieser Befehl dem MS–DOS–Befehlsprozessor zur Ausführung übermittelt.

Wenn Sie jetzt den Verzeichniseintrag für die Datei INVEST.RPT sehen wollen, können Sie einen Teil des Zeilenpufferinhaltes verwenden. Drücken Sie dazu zuerst die Taste **F2** und dann **m**. Dadurch werden alle Zeichen aus dem Zeilenpuffer bis zum Buchstaben **m**, aber ohne diesen, in die Befehlszeile kopiert. Sie erhalten also die Anzeige

```
dir invest._
```

Der blinkende Unterstreichungsstrich am Ende der Befehlszeile ist der Cursor.

Geben Sie jetzt die Buchstaben **rpt** ein:

```
dir invest.rpt_
```

Wenn Sie dann die Eingabetaste drücken, wird der Befehl ausgeführt und die veränderte Befehlszeile wird zum neuen Inhalt des Zeilenpuffers.

Jetzt wollen Sie sich den Inhalt der Datei INVEST.RPT mit dem Befehl TYPE INVEST.RPT ansehen. Dazu gehen Sie folgendermaßen vor:

- Geben Sie ein: `type` ohne die Eingabetaste zu drücken.
- Drücken Sie die **Ins**–Taste.
- Drücken Sie die Leertaste.
- Drücken Sie die **F3**–Taste.
- Drücken Sie die Eingabetaste.

Während der Texteingabe erscheinen die den gedrückten Zeichentasten zugeordneten Buchstaben direkt auf dem Bildschirm. Gleichzeitig wird der Zeichenzeiger im Zeilenpuffer für jedes eingegebene Zeichen um eine Position nach rechts gesetzt. Vor dem Drücken der **Ins**–Taste steht der Zeiger im Puffer also hinter dem Leerzeichen, das auf DIR folgt, da TYPE vier Zeichen lang ist. Nach dem Drücken der **Ins**–Taste wird das zur Texteingabe synchrone Weitersetzen des Zeichenzeigers im Zeilenpuffer ausgeschaltet.

Um ein Leerzeichen zwischen dem Wort TYPE und der Dateiangabe INVEST.RPT einzufügen, wird die **Ins**– und dann die Leertaste gedrückt. Mit der Taste **F3** wird der Rest des Zeilenpuffers von der aktuellen Position des Zeichenzeigers bis zum Ende in die Befehlszeile kopiert. Drücken der Eingabetaste führt schließlich den Befehl aus und überträgt die neue Befehlszeile in den Zeilenpuffer, der damit folgendes enthält:

```
type invest.rpt
```

Wenn Sie sich z.B. bei dem Befehlswort TYPE verschrieben haben und statt dessen PYTE eingegeben haben, zeigt MS–DOS eine Fehlermeldung an. Dennoch brauchen Sie nicht den gesamten Befehl mit Parametern neu einzugeben, sondern Sie können zunächst die fehlerhafte Befehlszeile in den Zeilenpuffer übernehmen, indem Sie **anstelle** der Eingabetaste die **F5**–Taste drücken. Dann können Sie mit Hilfe der **F1**– und **F3**–Tasten sowie durch Eingeben der richtigen Zeichen an den richtigen Stellen die Befehlszeile korrigieren. Also

```
pyte invest.rpt_
```

jetzt **F5** drücken:

```
pyte invest.rpt@
_
```

Dann Befehlszeile mit **F1**– und **F3**–Taste korrigieren:

```
pyte invest.rpt@
type invest.rpt
```

Alternativ können Sie auch ausgehend vom falschen Zeilenpufferinhalt PYTE INVEST.RPT durch Betätigen der folgenden Tasten zum selben Ergebnis kommen:

Del Del F1 Ins yp F3

Im einzelnen passiert dabei folgendes:

Del _	das erste Zeichen im Zeilenpuffer wird übergangen.
Del _	das zweite Zeichen im Zeilenpuffer wird übergangen.
F1 t_	das dritte Zeichen aus dem Zeilenpuffer wird in die Befehlszeile kopiert
Ins yp typ_	die beiden Zeichen y und p werden eingefügt
F3 type invest.rpt	der Rest des Zeilenpuffers wird in die Befehlszeile kopiert.

Beachten Sie hier, daß das Betätigen der **Del**–Taste nicht auf die Befehlszeile wirkt, sondern statt dessen ein Zeichen im Zeilenpuffer löscht.

8. Der Zeileneditor EDLIN

In diesem Kapitel lernen Sie,

- wie Sie den Zeileneditor EDLIN unter MS–DOS starten;
- wie Sie den Zeileneditor EDLIN beenden;
- wie Sie die MS–DOS–Edier– und Funktionstasten bei EDLIN benutzen können;
- was jeder EDLIN–Befehl im einzelnen bewirkt;
- was die EDLIN–Fehlermeldungen bedeuten.

8.1 Einführung

Mit dem MS–DOS–Zeileneditor EDLIN können Sie folgende Arbeiten erledigen:

- Textdateien erzeugen und auf Disk speichern.
- Existierende Textdateien verändern und speichern, ggf. unter Erhaltung der Originaldatei.
- Zeilen aus Textdateien löschen, edieren, einfügen oder anzeigen.
- Text in einer Textdatei suchen, löschen oder ersetzen.
- Notizen, Briefe, Berichte oder Quellsprachenprogramme für Assembler, GW–BASIC–Interpreter oder Compiler erstellen.

EDLIN unterteilt den Text aus einer Datei in Zeilen, von denen jede bis zu 253 Zeichen lang sein darf. Jeder Zeile wird eine fortlaufende Nummer zugeteilt, die auf dem Bildschirm angezeigt wird, **nicht** jedoch Bestandteil der Textdaten ist.

Wenn Textzeilen eingefügt oder gelöscht werden, werden die folgenden Zeilen automatisch renumeriert.

8.2 EDLIN starten

Das allgemeine Format von Edlin ist das folgende:

[*L*:][*BPfad*]**EDLIN** [*Laufw*:][*Pfad*]*Datei*[*.Erw*] [**/B**]

Es bedeuten:

L:BPfad Laufwerk und Suchpfad für die EDLIN–Programmdatei. Werden diese Angaben weggelassen, so sucht MS–DOS auf dem voreingestellten Suchpfad nach der EDLIN–Programmdatei.

Laufw:PfadDatei.Erw Die vollständige Angabe für die Textdatei. Wird ***Laufw:*** weggelassen, so nimmt EDLIN das aktuelle Laufwerk an. Wird ***Pfad*** weggelassen, so nimmt EDLIN das aktuelle Verzeichnis an. Wird ***.Erw*** weggelassen, so vergibt EDLIN auch keine Namenserweiterung. EDLIN sucht die angegebene Datei. Wird sie nicht gefunden, so wird dies in einer Meldung angezeigt und EDLIN erzeugt diese Datei.

B/ Wird dieser Schalter eingegeben, so betrachtet EDLIN ggf. in der Datei vorkommende Ctrl–Z–Zeichen **nicht** als Datei-Endmarke.

Angenommen, Sie wollen eine Datei BUDGET.JUN im aktuellen Verzeichnis der Disk im aktuellen Laufwerk mit EDLIN erzeugen. Dazu müssen Sie eingeben:

```
edlin budget.jun
```

Sie erhalten dann eine Anzeige, daß es sich um eine neue Datei handelt:

```
New file
*_
```

Der Stern ist die Bereitschaftsanzeige von EDLIN.

Um mit der Texteingabe beginnen zu können, müssen Sie den Befehl **I** (**i**nsert = einfügen; s. dort in Kapitel 8.8) eingeben.

Beachten Sie: Die Eingabe von Befehlen oder Text wird grundsätzlich durch Drücken der Eingabetaste beendet.

Wenn Sie dagegen beispielweise die existierende Datei BUDGET.MAI bearbeiten wollen, geben Sie zunächst auch ein:

```
edlin budget.mai
```

Wenn EDLIN diese Datei gefunden hat, wird diese in den Speicher geladen. Paßt sie ganz in den Speicher, meldet EDLIN

```
End of input file
*_
```

(Ende der Eingabedatei). Anschließend können Sie diese Datei mit den EDLIN–Befehlen (s. Kapitel 8.8) bearbeiten.

Ist die Datei dagegen für den verfügbaren Speicher zu groß, lädt EDLIN den Speicher etwa 3/4 voll und meldet sich dann mit seiner Bereitschaftsanzeige. Jetzt können Sie zunächst den im Speicher verfügbaren Teil edieren. Dann speichern Sie diesen Teil auf Disk und können anschließend den Rest der Datei laden (s. **W**– und **A**–Befehle in Kapitel 8.8).

8.3 EDLIN beenden

Am Ende Ihrer Arbeit mit EDLIN können Sie die Originaldatei wie auch die bearbeitete Textdatei mit dem **E**–Befehl (s. dort in Kapitel 8.8) auf Disk speichern. Der Originaldatei weist EDLIN die Namenserweiterung .BAK zu. Die edierte Datei erhält den beim Aufruf von EDLIN angegebenen Namen.

Achtung: Sie können eine Datei mit der Erweiterung .BAK nicht mit EDLIN bearbeiten. EDLIN sichert nämlich grundsätzlich die Originaldatei unter dem Namen .BAK, so daß Sie Ihre edierten Daten verlieren würden. Müssen Sie eine solche Datei edieren, so benennen Sie sie vorher mit dem MS–DOS–Befehl REN (s. dort in Kapitel 6.3.1) um und verwenden Sie dann diesen Namen beim Aufruf von EDLIN.

8.4 Spezielle Ediertasten

Während der Texteingabe oder der Bearbeitung einer Textzeile können die schon in Kapitel 7 erwähnten speziellen Edier– und Funktionstasten benutzt werden, da auch EDLIN die zuletzt eingegebene Zeile zusätzlich in einem Eingabepuffer zwischenspeichert. Im einzelnen sind dies die Tasten:

F1 **oder** 6 → kopiert ein Zeichen aus dem Eingabepuffer in die aktuelle Eingabzeile.

F2	kopiert alle Zeichen aus dem Eingabepuffer bis zu einem nach dem Drücken der F2–Taste eingegebenen Zeichen in die aktuelle Eingabezeile. Dieses Zeichen wird nicht mehr mit angezeigt.
F3	kopiert alle Zeichen aus dem Eingabepuffer von der Position des Eingabepufferzeigers an in die aktuelle Eingabezeile.
Del	löscht das Zeichen im Eingabepuffer bei der aktuellen Position des Eingabepufferzeigers.
F4	löscht alle Zeichen im Eingabepuffer bis zu einem nach dem Drücken der F4–Taste eingegebenen Zeichen. Das eingegebene Zeichen wird nicht mit gelöscht.
Esc	löscht die aktuelle Eingabezeile auf dem Bildschirm. Der Inhalt des Eingabepuffers bleibt unverändert.
Ins	schaltet den Einfügemodus ein oder aus. Bei eingeschaltetem Einfügemodus wird die aktuelle Eingabezeile auf dem Bildschirm um die eingegebenen Zeichen ergänzt, ohne daß der Zeichenzeiger im Eingabepuffer entsprechend weitergestellt wird.
F5	übernimmt die aktuelle Eingabezeile vom Bildschirm in den Eingabepuffer.

Im folgenden wird jede der o.a. Funktions–Tasten ausführlich beschrieben.

F1 oder →

Bedeutung: Kopiert ein Zeichen aus dem Zeilenpuffer in die aktuelle Eingabezeile.

Beispiel: Geben Sie, nachdem Sie EDLIN mit einem beliebigen Dateinamen gestartet haben, folgendes ein und drücken am Zeilenende die Eingabetaste:

```
1:*Bilanzbericht 1985
2:*_
```

Nach dem Drücken der Eingabetaste befindet sich der Cursor (_) am Anfang der zweiten Zeile. Wenn Sie jetzt einmal die **F1**– oder →–Taste drücken, kopiert EDLIN das erste Zeichen aus dem Eingabepuffer in die aktuelle Eingabezeile 2:

```
F1
2:*B_
```

Jedes weitere Drücken holt weitere Zeichen aus dem Eingabepuffer:

```
F1
2:*Bi_

F1
2:*Bil_

F1
2:*Bila_
```

Anmerkungen:

- Drücken der **F1**– oder →–Taste schaltet den Einfügemodus (s.a. **Ins**–Taste) aus.

F2

Bedeutung: kopiert alle Zeichen aus dem Eingabepuffer bis zu einem nach dem Drücken der F2–Taste eingegebenen Zeichen in die aktuelle Eingabezeile. Dieses Zeichen wird nicht mehr mit angezeigt.

Beispiel: Geben Sie, nachdem Sie EDLIN mit einem beliebigen Dateinamen gestartet haben, folgendes ein und drücken am Zeilenende die Eingabetaste:

```
1:*Bilanzbericht 1985
2:*_
```

Nach dem Drücken der Eingabetaste befindet sich der Cursor (_) am Anfang der zweiten Zeile. Wenn Sie jetzt einmal die **F2**– und dann die Taste **C** drücken, kopiert EDLIN den Inhalt des Zeilenpuffers bis ausschließlich zum Buchstaben **c** in die aktuelle Eingabezeile:

```
F2c
2:*Bilanzberi_
```

Anmerkungen:

- EDLIN kopiert hier immer von der aktuellen Position des Eingabepufferzeigers bis zum angegebenen Zeichen. Ist das angegebene Zeichen im verbleibenden Pufferteil nicht mehr vorhanden, so wird auch nichts kopiert.

- Drücken der **F2**–Taste schaltet den Einfügemodus (s.a. **Ins**–Taste) aus.

F3

Bedeutung: kopiert alle Zeichen aus dem Eingabepuffer von der Position des Eingabepufferzeigers an in die aktuelle Eingabezeile.

Beispiel: Geben Sie, nachdem Sie EDLIN mit einem beliebigen Dateinamen gestartet haben, folgendes ein (in der zweiten Zeile drücken Sie nach dem Wort **Jahres** **nicht** die Eingabetaste):

```
1:*Bilanzbericht 1985
2:*Jahres_
```

Wenn Sie jetzt einmal die **F3**–Taste drücken, erhalten Sie:

```
1:*Bilanzbericht 1985
2:*Jahresbericht 1985_
```

Anmerkungen:

- Drücken der **F3**–Taste schaltet den Einfügemodus (s.a. **Ins**–Taste) aus.

Del

Bedeutung: löscht das Zeichen im Eingabepuffer bei der aktuellen Position des Eingabepufferzeigers.

Beispiel: Geben Sie, nachdem Sie EDLIN mit einem beliebigen Dateinamen gestartet haben, folgendes ein und drücken am Zeilenende die Eingabetaste:

```
1:*Bilanzbericht 1985
2:*_
```

Nach dem Drücken der Eingabetaste befindet sich der Cursor (_) am Anfang der zweiten Zeile. Wenn Sie jetzt einmal die **Del**–Taste drücken, erhalten Sie

```
Del
2:*_
```

Der Cursor bleibt an seiner alten Position. Die **Del**–Taste wirkt also nur auf den Eingabepuffer. Sie können dies überprüfen, indem Sie jetzt die **F3**–Taste drücken:

```
F3
ilanzbericht 1985_
```

F4

Bedeutung: löscht alle Zeichen im Eingabepuffer bis zu einem nach dem Drücken der F4–Taste eingegebenen Zeichen. Das eingegebene Zeichen wird nicht mit gelöscht.

Beispiel: Geben Sie, nachdem Sie EDLIN mit einem beliebigen Dateinamen gestartet haben, folgendes ein und drücken am Zeilenende die Eingabetaste:

```
1:*Bilanzbericht 1985
2:*_
```

Nach dem Drücken der Eingabetaste befindet sich der Cursor (_) am Anfang der zweiten Zeile. Wenn Sie jetzt einmal die **F4**– und dann die Taste **C** drücken, erhalten Sie folgende Anzeige:

```
F4c
2:*_
```

Der Cursor bleibt an seiner alten Position. Die **F4**–Taste wirkt also nur auf den Eingabepuffer. Sie können dies überprüfen, indem Sie jetzt die **F3**–Taste drücken:

```
F3
cht 1985_
```

Esc

Bedeutung: Löscht die aktuelle Eingabezeile auf dem Bildschirm. Der Inhalt des Eingabepuffers bleibt unverändert.

Beispiel: Geben Sie, nachdem Sie EDLIN mit einem beliebigen Dateinamen gestartet haben, folgendes ein und drücken am Zeilenende der ersten Zeile die Eingabetaste:

```
1:*Bilanzbericht 1985
2:*Firma Commodore_
```

Um die aktuelle Zeile (Zeile 2) zu löschen, drücken Sie jetzt die **Esc**–Taste:

```
Esc
2:*Firma Commodore\
_
```

Der linke Schrägstrich am Ende der Zeile markiert diese Zeile als gelöscht. Drücken Sie jetzt die Eingabetaste, um mit der Arbeit fortzufahren. Daß der Eingabepuffer unverändert geblieben ist, können Sie durch Drücken der **F3**–Taste überprüfen:

```
F3
3:*Bilanzbericht 1985
```

Anmerkungen:

- **Esc** markiert die gelöscht Zeile durch einen linken Schrägstrich (\) und setzt den Cursor auf die nächste Bildschirmzeile, die damit aktuelle Zeile wird.

- Drücken der **Esc**–Taste schaltet den Einfügemodus (s.a. **Ins**–Taste) aus.

Ins

Bedeutung: Schaltet den Einfügemodus ein oder aus. Bei eingeschaltetem Einfügemodus wird die aktuelle Eingabezeile auf dem Bildschirm um die eingegebenen Zeichen ergänzt, ohne daß der Zeichenzeiger im Eingabepuffer entsprechend weitergestellt wird.

Beispiel: Geben Sie, nachdem Sie EDLIN mit einem beliebigen Dateinamen gestartet haben, folgendes ein und drücken am Zeilenende die Eingabetaste:

```
1:*Bilanzbericht 1985
2:*_
```

Nach dem Drücken der Eingabetaste befindet sich der Cursor (_) am Anfang der zweiten Zeile. Jetzt drücken Sie die beiden Tasten **F2** und **B**:

F2b

```
2:*Bilanz_
```

Drücken Sie dann die **Ins**–Taste und geben Sie den Text **– und Jahres** ein, ohne die Eingabetaste zu drücken:

Ins

```
2:*Bilanz- und Jahres_
```

Wenn Sie jetzt die **F3**–Taste drücken, erhalten Sie schließlich:

F3

```
2:*Bilanz- und Jahresbericht 1985_
```

Wenn Sie die Eingabetaste nach dem Einschalten des Einfügemodus' drücken, kopiert EDLIN nicht den Rest des Eingabepuffers in die aktuelle Eingabezeile, sondern schließt diese nach dem eingefügten Text ab:

```
1:*Bilanzbericht 1985
2:*Bilanz- und Jahres
3:*_
```

Um den Einfügemodus wieder abzuschalten (und damit wieder in den voreingestellten Überschreibmodus zu gelangen), brauchen Sie nur die **Ins**–Taste ein zweites Mal zu drücken. Danach überschreiben wieder alle eingegebenen Zeichen die an der entsprechenden Position im Eingabepuffer stehenden Zeichen.

Für ein weiteres Beispiel wird wieder von der ursprünglichen Eingabe ausgegangen:

```
1:*Bilanzbericht 1985
2:*_
```

Geben Sie jetzt die im folgenden angegebenen Tastenfolgen ein. Das Ergebnis ist jeweils in der Folgezeile angegeben.

F2 t
```
2:*Bilanzberich
```

Del Ins te 1984
```
2:*Bilanzberichte 1984 _
```

Ins und 1985 Eingabetaste
```
2:*Bilanzberichte 1984 und 1985
3:*_
```

Sie haben **te 1984** eingefügt und **und 1985** ersetzt. Der Eingabepuffer enthält jetzt:

```
Bilanzberichte 1984 und 1985
```

Anmerkungen:

- Nach dem Start befindet sich EDLIN automatisch im Überschreib–Modus, d.h. jedes eingegebene Zeichen stellt den Eingabepufferzeiger eine Position nach rechts. Wenn einmal die **Ins**–Taste gedrückt wird, wird der Einfügemodus eingeschaltet. Danach wird bei Zeicheneingabe der Eingabpufferzeiger **nicht** mehr weitergestellt, wohl aber der Cursor in der aktuellen Eingabezeile auf dem Bildschirm. Wenn am Ende der Einfügung die **Ins**–Taste noch einmal gedrückt wird, wird wieder der Überschreibmodus mit der o.a. Wirkung auf den Eingabepuffer eingeschaltet.

F5

Bedeutung: Übernimmt die aktuelle Zeile vom Bildschirm in den Eingabepuffer und löscht dabei dessen vorherigen Inhalt.

Beispiel: Geben Sie, nachdem Sie EDLIN mit einem beliebigen Dateinamen gestartet haben, folgendes ein und drücken am Zeilenende die Eingabetaste:

```
1:*Bilanzbericht 1985
2:*_
```

Nach dem Drücken der Eingabetaste befindet sich der Cursor (_) am Anfang der zweiten Zeile. Geben Sie jetzt die im folgenden angegebenen Tastenfolgen ein. Das Ergebnis ist jeweils in der Folgezeile angegeben.

F2 `t`
```
2:*Bilanzberich
```

Del **Ins** `te 1984`
```
2:*Bilanzberichte 1984 _
```

Ins `und 1985`
```
2:*Bilanzberichte 1984 und 1985_
```

Nehmen Sie an, Sie wollen jetzt am Anfang der 2. Zeile das Wort `Die` einfügen, ohne aber die ganze Zeile noch einmal schreiben zu müssen. Dazu gehen Sie wie folgt vor:
Drücken Sie zunächst die **F5**–Taste

F5
```
2:*Bilanzberichte 1984 und 1985@
_
```

Das @–Zeichen am Ende zeigt an, daß diese Zeile jetzt auch im Eingabepuffer steht. Um jetzt das Wort `Die`, gefolgt von einer Leerstelle, am Zeilenanfang einzufügen, drücken Sie die **Ins**–Taste und geben dann die

folgenden Tastenfolgen ein. Das Ergebnis steht in der jeweils folgenden Zeile:

Ins `Die`
`2:*Die _`

F3
`2:*Die Bilanzberichte 1984 und 1985`

Anmerkungen:

- **F5** markiert die bezogene Zeile am Ende mit dem @–Zeichen und setzt den Cursor auf den Anfang der nächsten Zeile.

- den Einfüge–Modus (s.a. **Ins**–Taste) aus.

- **F5** hat dieselbe Funktion wie die **Esc**–Taste, mit der Ausnahme, daß am Zeilenende ein @– anstelle eines \–Zeichens angefügt wird und daß der Inhalt des Eingabepuffers verändert wird.

8.5 Die EDLIN–Befehle: Eine Übersicht

Neben den im letzten Abschnitt beschriebenen Möglichkeiten, die aktuelle und die davor zuletzt eingegebene Textzeile mit Hilfe der Edier– und Funktionstasten zu manipulieren, gibt es eine ganze Reihe von Befehlen, mit denen Textzeilen oder ganze Textpassagen verändert oder kopiert sowie gespeichert oder geladen werden können. EDLIN kennt folgende Befehle:

Befehl	Name	Wirkung
A	Append	fügt Textzeilen an.
C	Copy	kopiert Textzeilen.
D	Delete	löscht Textzeilen.
Zeile	Edit	ediert eine oder mehrere Zeilen.
E	End	beendet die Arbeit mit EDLIN.
I	Insert	fügt Textzeilen ein.
L	List	zeigt Textzeilen an.
M	Move	verschiebt einen Zeilenblock.
P	Page	blättert bildschirmweise durch den Text.
Q	Quit	beendet die Arbeit mit Edlin, ohne den Text auf Disk zu speichern.
R	Replace	ersetzt Text.
S	Search	sucht einen angegebenen Text.
T	Transfer	lädt Text aus einer anderen Datei zum edierten Text dazu.
W	Write	schreibt einen Textblock auf Disk.

8.6 Hinweise zur Benutzung der EDLIN–Bfehle

Nach dem Start von EDLIN mit einer bereits existierenden Textdatei können Sie mit Hilfe der EDLIN–Befehle diesen Text manipulieren. Im folgenden finden Sie einige Hinweise, die Sie bei der Benutzung der EDLIN–Befehle beachten sollten:

- Beim Aufruf von EDLIN sind Pfadnamen für den Suchpfad der zu edierenden Datei erlaubt. So können Sie die Datei OKTOBER im Unterverzeichnis \MARIA\TERMINE durch die Angabe von

  ```
  edlin \maria\termine\oktober
  ```

  edieren.

- Sie können sich auf Textzeilen–Nummern relativ zur aktuellen Zeile, die EDLIN durch einen Stern (*) kennzeichnet, beziehen. Zeilen **vor** der aktuellen Zeile werden durch ihre Nummer mit vorangestelltem Minus-Zeichen (–), **nach** der aktuellen Zeile durch ein Plus–Zeichen (+) bezeichnet. So listet z.B. der folgende Befehl 10 Zeilen vor der aktuellen Zeile, die aktuelle Zeile selbst sowie 10 Zeilen nach der aktuellen Zeile:

  ```
  -10,+10L
  ```

  Beachten Sie: Die Befehlsbuchstaben können sowohl als Klein– als auch als Großbuchstabe angegeben werden.

- Zwischen einer ggf. spezifizierten Zeilennummer und dem Befehlsbuchstaben kann eine Leerstelle angegeben werden, muß aber nicht. So sind folgende Befehle zum Löschen von sechs Zeilen identisch:

  ```
  6d
  6 d
  6D
  6 D
  ```

- Es dürfen mehrere Befehle in einer Befehlszeile stehen, indem sie einfach hintereinander eingegeben werden.

  Wenn jedoch darunter der Befehl ***Zeile*** **edieren** (Edit–Befehl) ist, muß die Zeilenangabe mit einem Semikolon (;) vom nachfolgenden Befehl getrennt werden.

  Am Ende einer Zeichenkette beim Such–Befehl (**S**) oder beim Ersetz-Befehl (**R**) muß **Ctrl–Z** oder **F6** anstelle der Eingabetaste gedrückt werden. Bei **Ctrl**–Steuerzeichen wird die **Ctrl**–Taste niedergehalten und dann die angegebene Buchstabentaste gedrückt. Diese Tastenkombinationen sind bei den folgenden Beispielen in **halbfetter** Schrift angegeben, und nicht in der `Bildschirmschrift`.

  Die folgende Befehlszeile ediert zunächst Zeile 15 und zeigt danach die Zeilen 10 bis 20 auf dem Bildschirm an:

  ```
  15;-5,+5L
  ```

  Die Befehlszeile im nächsten Beispiel sucht nach der Zeile mit dem Text `Monatsbudget` und zeigt dann den Textblock bestehend aus 5 Zeilen vor der gesuchten Zeile, diese Zeile selbst und 5 Zeilen danach an:

```
SMonatsbudgetCtrl-Z-5,+5L
```

Beachten Sie: Wenn Sie einen Befehl mit einer zu suchenden Zeichenkette eingeben, so sucht EDLIN erst ab der aktuellen Zeile (die mit dem Stern (*)).

- Steuerzeichen, wie z.B. **Ctrl–C** können mit Hilfe von **Ctrl–V** in den Text oder Suchzeichenketten eingefügt werden. Dazu gehen Sie zunächst in den Einfüge–Modus (**Ins**–Taste drücken), dann drücken Sie **Ctrl–V** und anschließend den Großbuchstaben, der das einzufügende Steuerzeichen bezeichnet. Hier ein Beispiel für Ctrl–C:

  ```
  1*Dies ist ein AbbruchCtrl-VC
  ```

  Das nächste Beispiel ersetzt in allen Zeilen nach der aktuellen Zeile das Steuerzeichen **Ctrl–Z** durch das Wort **Bleistift**.

  ```
  RCtrl-VZCtrl-Z Bleistift
  ```

  Durch die Folge

  Ctrl–V

  wird das Steuerzeichen **Ctrl–V** in einen Text eingefügt.

- Normalerweise bedeutet das Steuerzeichen **Ctrl–Z** für Edlin die Datei-Endmarke. Wenn jedoch solche Steuerzeichen in Ihrem Text vorkommen, müssen Sie EDLIN mitteilen, daß diese Steuerzeichen nicht die Bedeutung einer Date-Endmarke haben. Dies können Sie, wenn Sie EDLIN mit dem /B–Schalter aufrufen (s. Kapitel 8.2).

- Fehlermeldungen, mit denen Sie bei der Arbeit mit EDLIN konfrontiert werden, sind im Anhang A bei den allgemeinen Fehlermeldungen zusammengestellt.

8.7 Parameter zur Befehlserweiterung

Viele EDLIN–Befehle können mit zusätzlichen Parametern (Optionen) in ihrer Wirkung erweitert werden. EDLIN unterscheidet drei verschiedene Parameter:

Zeile: Hier können Sie eine Zeilennummer angeben. Mehrere Zeilennummern müssen durch Kommata (,) oder Leerstellen voneinander getrennt werden.

Zeilennummern können auf dreierlei Weise angegeben werden:

1. Als ganze Dezimalzahl zwischen 1 und 65529. Wird eine größere Zahl als Zeilen im Hauptspeicher verfügbar sind, angegeben, so bezieht sich die angegebene Zahl auf die Zeile nach der letzten Zeile im Hauptspeicher (s.a. bei 3. (Nummernsymbol)).

2. Als Punkt (.). Der Punkt bezieht sich auf die aktuelle Zeile, also die Zeile, die zuletzt ediert wurde. Diese Zeile ist nicht notwendigerweise mit der zuletzt angezeigten Zeile identisch. Die aktuelle Zeile wird durch einen Stern (*) zwischen der Zeilennummer und dem ersten Textzeichen gekennzeichnet.

3. Als Nummern–Symbol (#). Das Nummern–Symbol bezeichnet die Zeile hinter der letzten Textzeile im Hauptspeicher.

Drücken Sie die Eingabetaste ohne eine der o.a. Zeilennummern–Kennzeichen bei einem Befehl, so verwendet EDLIN, abhängig vom Befehl, einen voreingestellten Zeilennummernwert.

Fragezeichen (?): Dieser Parameter wird nur in Verbindung mit dem Such–(**S**) oder dem Ersetz–(**R**) Befehl verwendet. Wird er angegeben, so fährt EDLIN nach dem Auffinden einer Zeile mit dem gesuchten Text erst fort, nachdem die Anforderung

```
O.K.?
```

durch Drücken der **Y**–Taste (Yes=Ja) bestätigt oder durch eine andere Taste für Nein verneint wird.

Text: Ein Text–Parameter gibt den Text an, nach dem gesucht, der ersetzt werden soll oder der anderen Text ersetzen soll. Auch dieser Parameter wird nur in Verbindung mit dem Such–(**S**) oder dem Ersetz–(**R**) Befehl benutzt. Jeder Suchtext muß mit **Ctrl–Z** oder der Eingabetaste abgeschlossen werden. Nur wenn Leerstellen Teil des gesuchten Textes sind, dürfen sie zwischen zwei Texten oder zwischen dem Befehls–Buchstaben und dem Text angegeben werden.

8.8 Die EDLIN–Befehle im einzelnen

Im folgenden wird jeder EDLIN–Befehl detailliert beschrieben. Das Format und die Syntax der Beschreibungen entsprechen denen bei der Beschreibung der MS–DOS–Befehle (s.a. Kapitel 6.2). Jede einzelne Befehlsbeschreibung beginnt auf einer neuen Seite.

A (Zeilen anfügen)

Format: [*n*]**A**

Bedeutung: Fügt weitere Textzeilen aus der edierten Datei zu dem im Hauptspeicher befindlichen Text hinzu.

n Eine ganze Zahl zwischen 1 und 65529 für die Anzahl der anzufügenden Zeilen. Wird ***n*** weggelassen, so fügt EDLIN solange Zeilen hinzu, bis der Hauptspeicher zu 3/4 gefüllt ist. Ist er bereits so voll, so wird der **A**–Befehl ignoriert.

Anmerkungen:

- Dieser Befehl wird nur benötigt, wenn die zu edierende Datei nicht auf einmal in den Hauptspeicher paßt. EDLIN liest beim Start immer soviele Zeilen in den Hauptspeicher, wie vorhanden sind oder bis der Speicher zu 3/4 gefüllt ist. Um den Rest der Datei zu edieren, müssen zunächst die bereits bearbeiteten Zeilen mit dem **W**–Befehl auf Disk gespeichert werden, ehe mit dem **A**–Befehl weitere Zeilen dazugeladen werden können.

- Wenn mit dem **A**–Befehl die letzte Zeile aus der Datei gelesen wurde, zeigt EDLIN die Meldung

  ```
  End of input file
  ```

  (Ende der Eingabedatei) an.

C (Zeilenblock kopieren)

Format: [*Z1*][,*Z2*],*Z3*[,*Anzahl*]**C**

Bedeutung: Kopiert einen angegebenen Zeilenblock ggf. wiederholt an eine angebene Position.

Z1, Z2 Zwei Zeilennummern, die die erste und letzte Zeile des zu kopierenden Zeilenblockes bezeichnen. Wird ***Z1*** weggelassen, so besteht der Block aus den Zeilen von der aktuellen Zeile bis zur Zeile ***Z2***. Wird ***Z2*** weggelassen, so wird nur ***Z1*** kopiert. Werden beide Parameter weggelassen, so wird nur die aktuelle Zeile kopiert.

Z3 Eine Zeilennummer, die die Zeile bezeichnet, **vor** die der Zeilenblock kopiert werden soll.

Anzahl Eine ganze Zahl, die angibt, wie oft der angegebene Zeilenblock kopiert werden soll. Wird ***Anzahl*** weggelassen, so wird der Zeilenblock einmal kopiert.

Anmerkungen:

- Liegt ***Z3*** zwischen ***Z1*** und ***Z2***, so gibt EDLIN eine Fehlermeldung aus.
- Die Textzeilen werden nach dem Kopieren neu durchnumeriert.

D (Zeilenblock löschen)

Format: [*Z1*][,*Z2*]**D**

Bedeutung: Löscht den angegebenen Zeilenblock und renumeriert die Datei im Speicher.

Z1, Z2 Zwei Zeilennummern, die die erste und letzte Zeile des zu löschenden Zeilenblockes bezeichnen. Wird ***Z1*** weggelassen, so besteht der Block aus den Zeilen von der aktuellen Zeile bis zur Zeile ***Z2***. Wird ***Z2*** weggelassen, so wird nur ***Z1*** gelöscht. Werden beide Parameter weggelassen, so wird nur die aktuelle Zeile gelöscht.

Beispiel: Folgende Datei werde ediert:

```
 1: Sehr geehrter Herr Schulze!
 2:
 3: Mit Bedauern habe ich von Ihrem letzten
 4: Unfall bei der Benutzung unseres auto-
 5: matischen Bleistiftanspitzers X-1000
 6: gehört.
 7:
 8: Ihr Unfall hat uns dazu veranlaßt,
 9: unsere Kunden künftig in der Bedie-
10: nungsanleitung vor dem Versuch, metal-
11: lene Stifte anzuspitzen, zu warnen.
12:
13:*Wir hoffen, daß Sie mit elektrischen
14: Geräten in Zukunft vorsichtiger um-
15: gehen werden!
16:
17: Mit freundlichem Gruß
18:
19:*Hans Spitzer, Direktor
```

Wenn Sie jetzt die Warnung an Herrn Schulze aus dem Brief streichen wollen, indem Sie die Zeilen 12 bis 15 löschen, müssen Sie folgendes eingeben:

```
12,15d
```

Als Ergebnis erhalten Sie:

```
 1: Sehr geehrter Herr Schulze!
 2:
 3: Mit Bedauern habe ich von Ihrem letzten
 4: Unfall bei der Benutzung unseres auto-
 5: matischen Bleistiftanspitzers X-1000
 6: gehört.
 7:
 8: Ihr Unfall hat uns dazu veranlaßt,
 9: unsere Kunden künftig in der Bedie-
10: nungsanleitung vor dem Versuch, metal-
11: lene Stifte anzuspitzen, zu warnen.
12:*
13: Mit freundlichem Gruß
14:
15: Hans Spitzer, Direktor
```

Um die Leerzeile 7 zu löschen, können Sie eingeben:

```
7d
```

Sie erhalten dann:

```
 1: Sehr geehrter Herr Schulze!
 2:
 3: Mit Bedauern habe ich von Ihrem letzten
 4: Unfall bei der Benutzung unseres auto-
 5: matischen Bleistiftanspitzers X-1000
 6: gehört.
 7:*Ihr Unfall hat uns dazu veranlaßt,
 8: unsere Kunden künftig in der Bedie-
 9: nungsanleitung vor dem Versuch, metal-
10: lene Stifte anzuspitzen, zu warnen.
11:
12: Mit freundlichem Gruß
13:
14: Hans Spitzer, Direktor
```

Aus diesem Brief können Sie jetzt eine kurze Anteilnahme machen, indem Sie die Zeilen 7 bis 11 löschen. Da Zeile 7 die aktuelle Zeile (mit dem Stern (*)) ist, brauchen Sie nur noch einzugeben:

```
,10d
```

Sie erhalten dann:

```
 1: Sehr geehrter Herr Schulze!
 2:
 3: Mit Bedauern habe ich von Ihrem letzten
 4: Unfall bei der Benutzung unseres auto-
 5: matischen Bleistiftanspitzers X-1000
 6: gehört.
 7:*
 8: Mit freundlichem Gruß
 9:
10: Hans Spitzer, Direktor
```

Beachten Sie bei diesen Beispielen, daß EDLIN nach dem Löschen die Datei immer wieder renumeriert, und daß immer die erste im D–Befehl angegebene Zeile nach dem Löschen die aktuelle Zeile wird.

Edit (Zeile edieren)

Format: [*Zeile*]

Bedeutung: Wird nur eine Zeilennummer eingegeben, so zeigt EDLIN diese Zeile zusammen mit der Nummer an und in der Zeile darunter noch einmal die Zeilennummer mit dem Cursor. Gleichzeitig wird der Inhalt dieser Zeile in den Eingapuffer kopiert. Jetzt kann die Zeile neu eingegeben, oder mit den Edier– und Funktionstasten (s.a. Kapitel 8.4) bearbeitet werden.

Zeile Eine ganze Zahl, die die Nummer der zu edierenden Zeile angibt. Wird ***Zeile*** nicht angegeben, sondern nur die Eingabetaste gedrückt, so zeigt EDLIN die auf die aktuelle Zeile folgende Zeile zum Edieren an.

Beispiel: Folgende Datei werde ediert:

```
1: Sehr geehrter Herr Schulze!
2:
3: Mit Bedauern habe ich von Ihrem letzten
4: Unfall bei der Benutzung unseres auto-
5: matischen Bleistiftanspitzers
6: gehört.
```

Nun soll in Zeile 5 der Produktname X–1000 eingefügt werden. Dazu müssen Sie zunächst

```
5
```

eingeben und die Eingabetaste drücken. EDLIN zeigt dann an:

```
5:*matischen Bleistiftanspitzers
5:*_
```

Jetzt kopieren Sie mit Hilfe der **F2**–Taste den Inhalt des Eingabepuffers bis zum Buchstaben B in die aktuelle Eingabezeile und geben dann den Produktnamen ein, also:

F2 B **Ins** X-1000-
```
5:*matischen X-1000-
```

F3 Eingabetaste
```
5:*matischen X-1000-Bleistiftanspitzers
```

Wenn Sie jetzt den Befehl **L** eingeben, erhalten Sie:

```
 1: Sehr geehrter Herr Schulze!
 2:
 3: Mit Bedauern habe ich von Ihrem letzten
 4: Unfall bei der Benutzung unseres auto-
 5:*matischen X-1000-Bleistiftanspitzers
 6: gehört.
```

Anmerkungen:

- Nach dem Drücken der Eingabetaste wird die edierte Zeile zur aktuellen Zeile.
- Soll die geänderte Zeile nicht gespeichert werden, so wird **Esc** oder **Ctrl–C** anstelle der Eingabetaste gedrückt. Die Originalzeile bleibt dann unverändert.
- Wird die Eingabetaste gedrückt, wenn der Cursor am Zeilenanfang steht, so hat dies dieselbe Wirkung wie Drücken von **Esc** oder **Ctrl–C**.
- Steht der Cursor an irgendeiner anderen Stelle als am Zeilenanfang oder –ende, so wird der Rest der Zeile beim Drücken der Eingabtaste gelöscht.

E (Text speichern und beenden)

Format: **E**

Bedeutung: Der gesamte im Hauptspeicher befindliche Text wird, nachdem die Originaldatei auf Disk mit der Namenserweiterung .BAK umbenannt wurde, in die Datei, die beim Aufruf von EDLIN angegeben wurde, gespeichert. War die angegebene Datei beim Aufruf von EDLIN noch nicht vorhanden, so wird auch keine .BAK–Datei erzeugt. Anschließend kehrt EDLIN auf die Befehlsebene von MS–DOS zurück.

Anmerkungen:

- Beim **E**–Befehl gibt es keine Befehlsparameter. D.h., die vollständige Dateiangabe muß bereits im Aufruf von EDLIN enthalten sein.

- Ehe Sie den **E**–Befehl geben, stellen Sie sicher, daß die Disk über ausreichend Speicherplatz verfügt. Andernfalls gehen alle Daten, die nicht mehr auf die Disk passen, verloren. In diesem Fall wird keine .BAK–Datei erzeugt und die unvollständige Datei erhält die Namenserweiterung .$$.

- Ist die letzte Textzeile der Datei nicht mit einer Wagenrücklauf–Zeilenvorschub–Folge abgeschlossen, so fügt EDLIN diese hinzu. Die Datei wird mit **Ctrl–Z** (1AH) als Datei-Endmarke abgeschlossen.

I (Zeilen einfügen)

Format: [*Zeile*]I

Bedeutung: Wenn Sie eine neue Datei edieren, müssen Sie am Anfang den I–Befehl geben, ehe Sie mit der Texteingabe beginnen können. In diesem Fall beginnt die Eingabe mit Zeile 1. Nach dem Drücken der Eingabetaste erscheint automatisch die nächste Zeilennummer. EDLIN bleibt solange im Einfügemodus, bis **Ctrl–C** gedrückt wird. Dann wird die Zeile, die auf die eingefügten Zeilen folgt, aktuelle Zeile (gekennzeichnet durch den Stern (*)).

Zeile Eine ganze Zahl, die die Nummer der Zeile bezeichnet, vor der Text eingegeben werden soll. Wird ***Zeile*** weggelassen, so wird vor der aktuellen Zeile eingefügt. Ist ***Zeile*** größer als die höchste vorhandene Zeilennummer, oder wird anstelle von ***Zeile*** das Nummern–Zeichen (#) eingegeben, so werden die Zeilen am Ende der Datei eingefügt. Auch in diesem Fall ist die aktuelle Zeile die Zeile, die auf die zuletzt eingegebene folgt, obwohl sie nicht existiert.

Beispiele: Folgende Datei werde ediert:

```
 1: Sehr geehrter Herr Schulze!
 2: .
 3: Mit Bedauern habe ich von Ihrem letzten
 4: Unfall bei der Benutzung unseres auto-
 5: matischen Bleistiftanspitzers X-1000
 6: gehört.
 7:
 8: Mit freundlichem Gruß
 9:
10: Hans Spitzer, Direktor
```

Da dieser kurze Brief keinen Ausdruck des Bedauerns enthält, soll eine kurze Notiz hinzugefügt werden. Dieser Text soll vor der Zeile 8 eingegeben werden. Dazu geben Sie ein:

```
8i
```

Als Ergebnis erhalten Sie

```
8:*_
```

Jetzt geben Sie folgende Zeilen ein und drücken am Ende jeder Zeile die Eingabetaste:

```
 8:*Ihr Unfall hat uns dazu veranlaßt,
 9:*unsere Kunden künftig in der Bedie-
10:*nungsanleitung vor dem Versuch, metal-
11:*lene Stifte anzuspitzen, zu warnen.
12:*_
```

Zur Beendigung des Einfügemodus drücken Sie **Ctrl–C** am Anfang der Zeile 12 und dann **L**, um die Datei zu listen. Sie erhalten:

```
 1: Sehr geehrter Herr Schulze!
 2:
 3: Mit Bedauern habe ich von Ihrem letzten
 4: Unfall bei der Benutzung unseres auto-
 5: matischen Bleistiftanspitzers X-1000
 6: gehört.
 7:
 8: Ihr Unfall hat uns dazu veranlaßt,
 9: unsere Kunden künftig in der Bedie-
10: nungsanleitung vor dem Versuch, metal-
11: lene Stifte anzuspitzen, zu warnen.
12:*Mit freundlichem Gruß
13:
14: Hans Spitzer, Direktor
```

Um jetzt noch eine Leerzeile vor der aktuellen Zeile (12) einzufügen, geben Sie nur den **I**–Befehl ein. Sie erhalten:

```
12:*_
```

Drücken Sie jetzt die Eingabetaste und anschließend **Ctrl–C**, um den Einfügemodus zu beenden. Dann geben Sie den **L**–Befehl:

```
 1: Sehr geehrter Herr Schulze!
 2:
 3: Mit Bedauern habe ich von Ihrem letzten
 4: Unfall bei der Benutzung unseres auto-
 5: matischen Bleistiftanspitzers X-1000
 6: gehört.
 7:
 8: Ihr Unfall hat uns dazu veranlaßt,
 9: unsere Kunden künftig in der Bedie-
10: nungsanleitung vor dem Versuch, metal-
11: lene Stifte anzuspitzen, zu warnen.
12:
13:*Mit freundlichem Gruß
14:
15: Hans Spitzer, Direktor
```

Um weitere Zeilen am Ende anzufügen, geben Sie den Befehl **16I**. Sie erhalten:

```
16:*_
```

Geben Sie jetzt folgende Zeilen ein:

```
16:*Spitzer Büro-Hilfsmittel
17:*Der Weltführer der Büro-Anspitztechnik
18:*Unser Motto: "Auch Sie sollten Spitze sein"
19:*_
```

Drücken Sie zur Beendigung des Einfügemodus am Anfang von Zeile 19 **Ctrl–C** und listen dann alle Zeilen mit dem Befehl **1,23L**:

```
 1: Sehr geehrter Herr Schulze!
 2:
 3: Mit Bedauern habe ich von Ihrem letzten
 4: Unfall bei der Benutzung unseres auto-
 5: matischen Bleistiftanspitzers X-1000
 6: gehört.
 7:
 8: Ihr Unfall hat uns dazu veranlaßt,
 9: unsere Kunden künftig in der Bedie-
10: nungsanleitung vor dem Versuch, metal-
11: lene Stifte anzuspitzen, zu warnen.
12:
13: Mit freundlichem Gruß
14:
15: Hans Spitzer, Direktor
16: Spitzer Büro-Hilfsmittel
17: Der Weltführer der Büro-Anspitztechnik
18: Unser Motto: "Auch Sie sollten Spitze sein"
```

L (Zeilen auflisten)

Format: [*Z1*][,*Z2*]L

Bedeutung: Listet den angegebenen Zeilenblock einschließlich der beiden angegebenen Zeilen.

Z1, Z2 Zwei Zeilennummern, die die erste und letzte Zeile des zu listenden Zeilenblockes bezeichnen. Wird ***Z1*** weggelassen, so beginnt das Listen 11 Zeilen vor der aktuellen Zeile und endet bei der Zeile ***Z2***. Wird ***Z2*** weggelassen, so werden beginnend mit ***Z1*** 23 Zeilen gelistet. Werden beide Parameter weggelassen, so werden 23 Zeilen, beginnend mit der 11. Zeile vor und endend mit der 11. Zeile nach der aktuellen Zeile gelistet.

Beispiele: Folgende Datei existiere und werde ediert:

```
 1: Sehr geehrter Herr Schulze!
 2:
 3: Mit Bedauern habe ich von Ihrem letzten
 4: Unfall bei der Benutzung unseres auto-
 5: matischen Bleistiftanspitzers X-1000
 6: gehört.
 .
 .
 .
15: Zusätzlich werden wir bei den neuen
16: Modellen einen Metalldetektor vorsehen.
 .
 .
 .
26: Hans Spitzer, Direktor
27: Spitzer Büro-Hilfsmittel
28: Der Weltführer der Büro-Anspitztechnik
29: Unser Motto: "Auch Sie sollten Spitze sein"
```

Um jetzt einen Zeilenblock ohne Bezug zur aktuellen Zeile zu listen, können sie z.B. eingeben:

```
2,6L
```

Sie erhalten dann:

```
 2:
 3: Mit Bedauern habe ich von Ihrem letzten
 4: Unfall bei der Benutzung unseres auto-
 5: matischen Bleistiftanspitzers X-1000
 6: gehört.
```

Wenn Sie einen Zeilenblock, der mit der aktuellen Zeile beginnt, listen wollen, geben Sie ein:

16,L oder **,L**

Sie erhalten dann:

```
16: Modellen einen Metalldetektor vorsehen.
 .
 .
 .
26: Hans Spitzer, Direktor
27: Spitzer Büro-Hilfsmittel
28: Der Weltführer der Büro-Anspitztechnik
29: Unser Motto: "Auch Sie sollten Spitze sein"
```

Um einen 23–Zeilenblock mit der aktuellen Zeile in der Mitte zu listen, brauchen Sie nur einzugeben:

L

Sie erhalten dann:

```
 .
 5: matischen Bleistiftanspitzers X-1000
 6: gehört.
 .
 .
 .
15: Zusätzlich werden wir bei den neuen
16:*Modellen einen Metalldetektor vorsehen.
 .
 .
 .
26: Hans Spitzer, Direktor
27: Spitzer Büro-Hilfsmittel
```

M (Zeilenblock verschieben)

Format: [*Z1*],[*Z2*],*Z3*M

Bedeutung: Verschiebt einen angegebenen Zeilenblock an eine angegebene Position und renumeriert danach die gesamte Datei.

Z1,Z2 Zwei Zeilennummern, die die erste und letzte Zeile des zu verschiebenden Zeilenblockes bezeichnen. Wird ***Z1*** weggelassen, so besteht der Block aus den Zeilen von der aktuellen Zeile bis zur Zeile ***Z2***. Wird ***Z2*** weggelassen, so wird nur ***Z1*** verschoben. Werden beide Parameter weggelassen, so wird nur die aktuelle Zeile verschoben. Vor ***Z2*** darf auch ein Plus–Zeichen (+) angegeben werden. Z.B. verschiebt der Befehl **,+25,100M** alle Zeilen von einschließlich der aktuellen Zeile bis einschließlich der 25. Zeile nach der aktuellen Zeile vor die Zeile 100.

Z3 Eine Zeilennummer, die die Zeile bezeichnet, vor die der Zeilenblock verschoben werden soll.

Beispiel: Folgende Datei werde ediert:

```
 1: Sehr geehrter Herr Schulze!
 2:
 3: Mit Bedauern habe ich von Ihrem letzten
 4: Unfall bei der Benutzung unseres auto-
 5: matischen Bleistiftanspitzers X-1000
 6: gehört.
 7:
 8: Ihr Unfall hat uns dazu veranlaßt,
 9: unsere Kunden künftig in der Bedie-
10: nungsanleitung vor dem Versuch, metal-
11: lene Stifte anzuspitzen, zu warnen.
12:
13: Mit freundlichem Gruß
14:
15: Hans Spitzer, Direktor
16: Spitzer Büro-Hilfsmittel
17: Der Weltführer der Büro-Anspitztechnik
18: Unser Motto: "Auch Sie sollten Spitze sein"
```

Angenommen, Sie wollen das Firmen–Motto an den Briefanfang stellen. Dazu müssen Sie die Zeilen 16 bis 18 vor die Zeile 1 verschieben. geben Sie also ein:

```
16,18,1M
```

Sie erhalten dann mit dem Befehl **1,20L**:

```
 1: Spitzer Büro-Hilfsmittel
 2: Der Weltführer der Büro-Anspitztechnik
 3: Unser Motto: "Auch Sie sollten Spitze sein"
 4: Sehr geehrter Herr Schulze!
 5:
 6: Mit Bedauern habe ich von Ihrem letzten
 7: Unfall bei der Benutzung unseres auto-
 8: matischen Bleistiftanspitzers X-1000
 9: gehört.
10:
11: Ihr Unfall hat uns dazu veranlaßt,
12: unsere Kunden künftig in der Bedie-
13: nungsanleitung vor dem Versuch, metal-
14: ene Stifte anzuspitzen, zu warnen.
15:
16: Mit freundlichem Gruß
17:
18: Hans Spitzer, Direktor
```

Anmerkungen:

- Liegt ***Z3*** zwischen ***Z1*** und ***Z2***, so gibt EDLIN eine Fehlermeldung aus.

P (seitenweise auflisten)

Format: [*Z1*][,*S*]P

Bedeutung: Listet eine Seite der Datei mit wählbarer Seitenlänge beginnend nach der angegebenen Zeile.

Z1 Die Nummer der Zeile, nach der aufgelistet werden soll. Wird ***Z1*** nicht angegeben, so wird mit der auf die aktuelle Zeile folgenden Zeile begonnen.

S Die Anzahl der Zeilen, aus denen die gelistete Seite bestehen soll. Wird ***S*** nicht angegeben, so werden 23 Zeilen gelistet.

Anmerkungen:

- Der **P**–Befehl unterscheidet sich insoweit vom **L**–Befehl, als hier nach dem Listen die letzte gelistete Zeile zur neuen aktuellen Zeile (mit Stern (*) gekennzeichnet) wird.

Q (EDLIN beenden, ohne zu speichern)

Format: **Q**

Bedeutung: Es werden keinerlei Daten gespeichert und EDLIN kehrt nach eine Bestätigung des Abbruchs durch den Benutzer auf die MS–DOS–Befehlsebene zurück.

Anmerkungen:

- EDLIN fordert vom Benutzer mit der Anfrage

  ```
  Abort edit (Y/N)? _
  ```

  (Edierung abbrechen (J/N)?) eine Bestätigung des Abbruches, um ungewollten Datenverlust zu vermeiden. Wird hier mit **Y** (Yes = Ja) geantwortet, so kehrt EDLIN auf die MS–DOS–Befehlsebene zurück, ohne die edierte Datei gesichert zu haben. Bei **N** (No = Nein) bleibt EDLIN aktiv und die Edierung kann fortgesetzt werden.

- Zur Beendigung von EDLIN mit Speichern der edierten Datei siehe **E**–Befehl.

R (Text ersetzen)

Format: [*Z1*][,*Z2*][?]R[*Text1*][*Ctrl–ZText2*]

Bedeutung: Jedes Auftreten von ***Text1*** im angegebenen Zeilenbereich wird durch ***Text2*** ersetzt.

Z1 Die Nummer der Zeile, nach der mit der Suche des zu ersetzenden Textes begonnen werden soll. Wird ***Z1*** weggelassen, so beginnt die Suche bei der auf die aktuelle Zeile folgenden Zeile.

Z2 Die Nummer der Zeile, die als letzte nach dem zu ersetzenden Text durchsucht werden soll. Wird ***Z2*** weggelassen, so endet die Suche mit der letzten Zeile im Speicher (Zeile #; s.a. Kapitel 8.7).

Werden ***Z1*** und ***Z2*** weggelassen, so geht die Suche von der auf die aktuelle Zeile folgenden bis zur letzten Zeile im Speicher.

? Wird das Fragezeichen angegeben, so fordert EDLIN nach jeder Zeile, die geändert wurde, mit der Anzeige

```
O.K.?
```

eine Bestätigung vom Benutzer (Y für Ja; jede andere Taste für Nein). Bei Nein wird die Änderung bei der betreffenden Zeile nicht durchgeführt.

Text1 Eine beliebige Zeichenfolge, nach der die Datei durchsucht wird, und die durch ***Text2*** (s.u.) ersetzt werden soll. EDLIN ersetzt in allen Zeilen des angegebenen Bereiches, in denen ***Text1*** vorkommt, diesen durch ***Text2***. Wird ***Text1*** nicht angegeben, arbeitet EDLIN mit dem ***Text1*** vom vorhergehenden **R**–Befehl. Wurde vorher noch kein **R**–Befehl angewendet, wird dieser Befehl ohne Aktion beendet.

Text2 Eine beliebige Zeichenfolge, die im angegebenen Zeilenbereich in allen Zeilen, die ***Text1*** (s.o.) enthalten,

diesen ersetzt. ***Text1*** und ***Text2*** müssen, wenn beide angegeben werden, durch Drücken von **Ctrl–Z** oder **F6** am Ende von ***Text1*** voneinander getrennt werden. Wird ***Text2*** nicht angegeben, so kann hinter ***Text1*** auch die Eingabetaste als Abschluß gedrückt werden. In diesem Fall wird in allen Zeilen des angegebenen Bereiches ***Text1*** durch nichts ersetzt, also gelöscht.

Beispiele:

Folgende Datei werde ediert:

```
 1: Sehr geehrter Herr Schulze!
 2:
 3: Mit Bedauern habe ich von Ihrem letzten
 4: Unfall bei der Benutzung unseres auto-
 5: matischen Bleistiftanspitzers X-1000
 6: gehört.
 7:
 8: Ihr Unfall hat uns dazu veranlaßt,
 9: unsere Kunden künftig in der Bedie-
10: nungsanleitung vor dem Versuch, metal-
11: lene Stifte anzuspitzen, zu warnen.
12:
13: Mit freundlichem Gruß
14:
15: Hans Spitzer, Direktor
```

Nun soll in allen Zeilen das Wort **uns** durch das Wort **mich** ersetzt werden. Der Befehl dazu lautet:

```
1,15r uns Ctrl–Z mich
```

Sie erhalten als Ergebnis:

```
 4: Unfall bei der Benutzung micheres auto-
 8: Ihr Unfall hat mich dazu veranlaßt,
 9:*michere Kunden künftig in der Bedie-
```

Die Änderungen in den Zeilen 4 und 9 sind unsinnig. Sie können verhindert werden, indem im o.a. Befehl der ?–Parameter verwendet wird, also:

```
1,15?r uns Ctrl–Z mich
```

EDLIN zeigt jetzt jede geänderte Zeile an und fordert eine Bestätigung der Änderung:

```
4: Unfall bei der Benutzung micheres auto-
   O.K.? n
8: Ihr Unfall hat mich dazu veranlaßt,
   O.K.? y
9: michere Kunden künftig in der Bedie-
   O.K.? n
   *_
```

Der Befehl **L** zeigt die richtige Ausführung der Änderungen:

```
 1: Sehr geehrter Herr Schulze!
 2:
 3: Mit Bedauern habe ich von Ihrem letzten
 4: Unfall bei der Benutzung unseres auto-
 5: matischen Bleistiftanspitzers X-1000
 6: gehört.
 7:
 8: Ihr Unfall hat mich dazu veranlaßt,
 9: unsere Kunden künftig in der Bedie-
10: nungsanleitung vor dem Versuch, metal-
11: lene Stifte anzuspitzen, zu warnen.
12:
13: Mit freundlichem Gruß
14:
15: Hans Spitzer, Direktor
```

S (Text suchen)

Format: [*Z1*][,*Z2*][?]S[*Text*]

Bedeutung: Das erste Auftreten von ***Text*** im angegebenen Zeilenbereich wird angezeigt und die Suche wird beendet, es sei denn, der **?**–Parameter wurde angegeben.

Z1 Die Nummer der Zeile, nach der mit der Suche des Textes begonnen werden soll. Wird ***Z1*** weggelassen, so beginnt die Suche bei der auf die aktuelle Zeile folgenden Zeile.

Z2 Die Nummer der Zeile, die als letzte nach dem Text durchsucht werden soll. Wird ***Z2*** weggelassen, so endet die Suche mit der letzten Zeile im Speicher (Zeile # ; s.a. Kapitel 8.7).

Werden ***Z1*** und ***Z2*** weggelassen, so geht die Suche von der auf die aktuelle Zeile folgenden bis zur letzten Zeile im Speicher.

? Wird das Fragezeichen angegeben, so fordert EDLIN nach jeder Zeile, in der der angegebene Text gefunden wurde, mit der Anzeige

```
O.K.?
```

eine Bestätigung der Fortsetzung der Suche vom Benutzer (Y für Ja; jede andere Taste für Nein). Bei N wird die Suche fortgesetzt, bei Y beendet.

Text Eine beliebige Zeichenkette, nach der die Datei durchsucht werden soll. Die Zeichenkette beginnt mit dem ersten Zeichen nach dem Befehlsbuchstaben **S** und endet mit dem Drücken der Eingabetaste. Wird ***Text*** nicht angegeben, so verwendet EDLIN den Suchtext des zuletzt benutzten **R**– oder **S**–Befehls. Wurde noch keiner dieser Befehle benutzt, so wird der **S**–Befehl ohne Aktion beendet.

Beispiele: Folgende Datei werde ediert:

```
 1: Sehr geehrter Herr Schulze!
 2:
 3: Mit Bedauern habe ich von Ihrem letzten
 4: Unfall bei der Benutzung unseres auto-
 5: matischen Bleistiftanspitzers X-1000
 6: gehört.
 7:
 8: Ihr Unfall hat uns dazu veranlaßt,
 9: unsere Kunden künftig in der Bedie-
10: nungsanleitung vor dem Versuch, metal-
11: lene Stifte anzuspitzen, zu warnen.
12:
13: Mit freundlichem Gruß
14:
15: Hans Spitzer, Direktor
```

Um nach dem ersten Auftreten der Silbe **zu** zu suchen, können Sie eingeben:

```
1,15S zu
```

Nach dem Drücken der Eingabetaste erhalten Sie:

```
 4: Unfall bei der Benutzung unseres auto-
```

Zum Auffinden der nächsten Zeile mit der Silbe **zu** können Sie den o.a. Befehl durch die Tastenfolge **Del F3 Eingabetaste** modifizieren und erhalten:

```
,15s zu
 8: Ihr Unfall hat uns dazu veranlaßt,
```

Wenn Sie bis zum Auffinden der richtigen Zeile suchen wollen, benutzen Sie den Befehl

```
1,?S zu
```

und erhalten z.B.:

```
 4: Unfall bei der Benutzung unseres auto-
    O.K.? n
 8: Ihr Unfall hat uns dazu veranlaßt,
    O.K.? n
11: lene Stifte anzuspitzen, zu warnen.
    O.K.? y
```

Zeile 11 sei die wirklich gesuchte. Hier wird die Suche durch Quittierung der Anfrage mit **y** (Yes = Ja) beendet.

Anmerkungen:

- Wird der angegebene Text im angegebenen Bereich nicht gefunden, so meldet EDLIN

  Not found

  (nicht gefunden).

T (Text dazuladen)

Format: [*Z*]T[*Laufw*:]*Datei*[.*Erw*]

Bedeutung: Lädt den Inhalt der angegebenen Datei aus dem aktuellen Verzeichnis der Disk im angegebenen Laufwerk zu dem im Hauptspeicher befindlichen Text **vor** die angegebene Zeile.

Z Eine ganze Zahl, die die Zeile des im Speicher befindlichen Textes bezeichnet, **vor** die der Text dazugeladen werden soll. Wird ***Z*** weggelassen, so nimmt EDLIN die aktuelle Zeile an.

***Laufw*:** Eine gültige Laufwerksbezeichnung nach MS–DOS–Konventionen. Wird ***Laufw*:** weggelassen, so nimmt EDLIN das aktuelle Laufwerk an.

Datei.Erw Der Name der Datei, die dazugeladen werden soll. Die Datei muß im aktuellen Verzeichnis der Disk im angegebenen Laufwerk vorhanden sein. Pfadnamen sind hier also nicht erlaubt. Wurde beim Aufruf von EDLIN ein Pfad angegeben (s.a. Kapitel 8.2), so ist das damit bezeichnete Verzeichnis für die Dauer der Arbeit mit EDLIN das aktuelle Verzeichnis der Disk im beim Aufruf angegebenen Laufwerk.

Beispiel: Um den Inhalt der Datei REPORT.TXT aus dem aktuellen Verzeichnis der Disk in Laufwerk A vor die Zeile 12 des im Speicher befindlichen Textes zu laden, müssen Sie eingeben:

```
12 t a:report.txt
```

W (Textblock speichern)

Format: [*n*]W

Bedeutung: Speichert die angegeben Anzahl Zeilen, beginnend mit Zeile 1 in der beim Aufruf von EDLIN angegebenen Datei.

n Eine ganze Zahl, die die Anzahl der zu speichernden Zeilen angibt. Wird ***n*** nicht angegeben, so werden solange Zeilen gespeichert, bis wieder 3/4 des gesamten Haupspeichers verfügbar sind. **W** arbeitet nicht, wenn der Hauptspeicher weniger als 3/4 gefüllt ist.

Anmerkungen:

- Dieser Befehl wird nur benötigt, wenn sehr große Dateien ediert werden, die nicht auf einmal in den Hauptspeicher passen. Beim Start von EDLIN mit einer großen Datei werden solange Zeilen in den Hauptspeicher geladen, bis dieser zu 3/4 gefüllt ist. Ehe der Rest der Datei ediert werden kann, müssen die edierten Zeilen mit dem **W**–Befehl auf Disk gespeichert werden, bevor neue Zeilen hinzugefügt werden können.

9. Dateivergleich mit FC

In diesem Kapitel lernen Sie:

- wie man Dateien zeilenweise miteinander vergleicht.
- wie man Dateien zeichenweise miteinander vergleicht.

9.1 Einführung

Es kann manchmal nützlich sein, zwei Disk–Dateien miteinander zu vergleichen. Z.B. können Kopien einer Datei mit dem Original verglichen werden, um festzustellen, welche Version aktueller ist. Für diesen Zweck befindet sich auf einer der beiden MS–DOS–Systemdisketten das Programm FC.EXE.

FC vergleicht die Inhalte zweier Dateien oder zweier Sätze von Dateien. Die Ausgabe der Unterschiede kann entweder auf dem Bildschirm oder in eine dritte Datei erfolgen. Es können sowohl reine Text– als auch reine Binär–Dateien verglichen werden. Der Vergleich kann

- Zeile für Zeile oder
- Byte für Byte

erfolgen.

Der zeilenweise Vergleich isoliert Zeilenblöcke, die sich bei beiden Dateien unterscheiden und zeigt diese Blöcke an. Dieser Vergleichstyp wird bei Text–Dateien (auch als ASCII–Dateien bezeichnet) verwendet.

Der byteweise Vergleich zeigt alle Bytes an, die sich in den verglichenen Dateien unterscheiden. Dieser Vergleichstyp wird bei Binär–Dateien (Programm–Dateien) verwendet.

9.2 Einschränkungen beim Vergleich

FC benutzt einen großen Speicherbereich (ausreichend für 100 Zeilen) als Pufferspeicher beim Textdateivergleich. Wenn die beiden zu vergleichenden Dateien größer sind, lädt FC zunächst soviel wie möglich in den Speicher und

beginnt den Vergleich. Findet FC bei diesem Teil keine Übereinstimmung, so wird die Meldung

```
resynch failed. Files are too different.
```

(Keine Übereinstimmung. Dateien sind zu verschieden) angezeigt und FC geht auf die MS–DOS–Befehlsebene zurück.

Binärdateien, für die der verfügbare Speicher nicht ausreicht, vergleicht FC vollständig, indem es nach dem Vergleich der ersten Teile weitere Teile von der Disk nachlädt. Die Ausgabe der Unterschiede erfolgt in gleicher Weise wie beim Vergleich zweier vollständig in den Speicher passenden Dateien.

9.3 Der Aufruf von FC

Die beiden Aufruf–Formate von FC sind

für ASCII–Dateivergleich:

[*L*:][*BPfad*]**FC** [**/A**][**/C**][**/L**][**LB** *n*][**/N**][**/W**][*/nnnn*]
[***Laufw***:][***Pfad***]***Datei***[***.Erw***]
[***Laufw***:][***Pfad***]***Datei***[***.Erw***]

Für Binär–Dateivergleich:

[*L*:][*BPfad*]**FC** [**/B**][*/nnnn*]
[***Laufw***:][***Pfad***]***Datei***[***.Erw***]
[***Laufw***:][***Pfad***]***Datei***[***.Erw***]

Dabei bedeuten im einzelnen:

L:BPfad Laufwerk und Suchpfad für die FC–Programmdatei.

Die diversen Schalter, gekennzeichnet durch einen vorangestellten Schrägstrich (/), werden mit ihrer Bedeutung im nächsten Abschnitt beschrieben.

Laufw:PfadDatei.Erw Zwei vollständige Dateiangaben für die beiden miteinander zu vergleichenden Dateien. Wird jeweils ***Laufw:*** nicht angegeben, so nimmt FC das aktuelle Laufwerk an. Wird jeweils ***Pfad*** nicht angegeben, so nimmt FC das aktuelle Verzeichnis der Disk im angegebenen Laufwerk an. FC vergleicht die mit der ersten Dateiangabe spezifizierte Datei oder den

Satz von Dateien mit der mit der zweiten Dateiangabe spezifizierten Datei oder dem Satz von Dateien und zeigt die Unterschiede an. Bei beiden Dateiangaben sind die Jokerzeichen * und ? für globale Dateinamen erlaubt.

Wenn Sie z.B. die beiden Text–Dateien MONATL.REP und VERKAUF.REP im aktuellen Verzeichnis der Disk im aktuellen Laufwerk miteinander vergleichen wollen, brauchen Sie nur einzugeben:

```
fc /a monatl.rep verkauf.rep
```

Wenn Sie zwei ganze Sätze von Dateien miteinander vergleichen wollen, können Sie Jokerzeichen in den Dateinamen verwenden. So können z.B. die beiden Benutzer RUDOLF und MARIA jeweils die drei Dateien REPORT1.TXT, REPORT2.TXT und REPORT3.TXT in ihren Unterverzeichnissen haben und wollen nun Unterschiede feststellen. Dann lautet der Aufruf von FC:

```
fc c:\benutzer\rudolf\report?.txt
\benutzer\maria\report?.txt
```

FC vergleicht dann zunächst die Datei REPORT1.TXT aus dem Unterverzeichnissen RUDOLF auf der Festplatte im Laufwerk C mit der Datei REPORT1.TXT aus dem Unterverzeichnis MARIA, danach die beiden Dateien REPORT2.TXT und schließlich REPORT3.TXT. Da bei der zweiten Dateiangabe kein Laufwerk spezifiziert wurde, nimmt FC hier das aktuelle Laufwerk an.

9.4 Die FC–Schalter

Beim Aufruf von FC können insgesamt 9 Schalter angegeben werden, um verschiedene Optionen beim Dateivergleich wählen zu können. Im folgenden wird die Wirkung jedes Schalters detailliert beschrieben.

/A Dieser Schalter bewirkt eine verkürzte Ausgabe beim ASCII–Datei–Vergleich. Anstatt alle nicht übereinstimmenden Zeilen anzuzeigen, wird jetzt nur die erste und letzte Zeile eines unterschiedlichen Textblockes angezeigt. Die dazwischen liegenden Zeilen werden durch eine Folge von drei Punkten (. . .) dargestellt.

/B Dieser Schalter bewirkt einen binären Dateivergleich. Die beiden Dateien oder Dateisätze werden byteweise miteinander verglichen, ohne einen

Synchronisationsversuch nach einem Byteunterschied. Die Unterschiede werden in der Form

```
xxxxxxxx yy zz
```

angezeigt. ***xxxxxxxx*** ist dabei die relative Adresse des nicht übereinstimmenden Bytepaares, gerechnet vom Dateianfang aus. Die Adressen beginnen bei 00000000. ***yy*** und ***zz*** sind die beiden nicht übereinstimmenden Bytes in hexadezimaler Codierung aus den beiden verglichenen Dateien. Wenn eine Datei weniger Daten enthält als die andere, wird eine entsrechende Meldung angezeigt. Wenn z.B. die beiden Dateien REPORT1.TXT und REPORT2.TXT verglichen werden und REPORT1.TXT kürzer ist, so meldet FC:

```
fc: report2.txt longer than report1.txt
```

(fc: report2.txt ist länger als report1.txt).

Der /B–Schalter wird voreingestellt, wenn .EXE–, .COM–, .SYS–, .OBJ–, .LIB.– oder .BIN–Dateien miteinander verglichen werden.

/C Wird dieser Schalter angegeben, so wird beim ASCII–Dateivergleich **nicht** zwischen Klein– und Groß–Buchstaben unterschieden. Alle Buchstaben in den Dateien werden dann als Großbuchstaben betrachtet. Z.B. sind dann die beiden folgenden Zeilen gleich:

```
Was für ein herrliches Wetter haben wir heute

was für ein HERRLICHES wetter haben wir heute
```

/L Mit dem /L–Schalter werden die beiden angegebenen Dateien oder Dateisätze im ASCII–Modus verglichen. Dieser Schalter wird voreinstellungsmäßig gesetzt, wenn Dateien verglichen werden, die **nicht** die Namenserweiterungen .EXE, .COM, .SYS, .OBJ, .LIB. oder .BIN haben.

LB *n* Dieser Schalter setzt die Größe des Zeilenpuffers auf ***n*** Zeilen. Voreingestellt sind 100 Zeilen. Dateien, die mehr als diese Anzahl sich unterscheidender Zeilen haben, führen zum Abbruch des FC–Programms.

/N Ist dieser Schalter gesetzt, so werden bei einem ASCII–Vergleich vor den angezeigten Zeilen Zeilennummern gesetzt.

/T Bei gesetztem /T–Schalter werden Tabulator–Zeichen nicht in die entsprechende Zahl von Leerstellen expandiert. Voreinstellungsmäßig werden Tabulatorzeichen als Anzahl Leerstellen bis zur nächsten Spaltenposition behandelt, die ein ganzzahliges Vielfaches von 8 ist.

/W Der /W–Schalter veranlaßt FC alle mehrfach aufeinanderfolgenden Leerstellen oder Tabulatorzeichen als jeweils eine Leerstelle zu behandeln. Beachten Sie jedoch, daß mit dem /W–Schalter Mehrfachleerstellen nur komprimiert werden; sie werden **nicht** ignoriert, es sei denn, es handelt sich um führende oder nachfolgende Leerstellen. Im folgenden Beispiel, bei dem die Leerstellen zur Demonstration durch das Zeichen _ dargestellt sind, werden die ersten drei Zeilen als gleich betrachtet, wenn der /W–Schalter gesetzt ist, die letzte dagegen nicht:

```
Keine_Daten_mehr_vorhanden
___Keine__Daten_mehr____vorhanden
__________Keine___________Daten_______mehr_vorhanden__
____KeineDaten_mehr_vorhanden
```

/nnnn Mit diesem Schalter wird die Anzahl ***nnnn*** von Zeilen angegeben, die bei den beiden verglichenen Dateien übereinstimmen müssen, ehe ein Unterschied auftreten darf. Enthalten die Dateien weniger übereinstimmende Zeilen, als diese Zahl angibt, so werden die übereinstimmenden Zeilen als unterschiedlich angezeigt.

9.5 Wie Dateiunterschiede angezeigt werden

Das Protokoll der Dateiunterschiede, das FC beim Vergleich erstellt, enthält bei jedem angezeigten Block zunächst den ersten beim Aufruf angegebenen Dateinamen gefolgt von den sich unterscheidenden Zeilen sowie der ersten wieder übereinstimmenden Zeile. Dann folgt der zweite beim Aufruf angegebene Dateiname, die gefundenen unterschiedlichen Zeilen sowie auch hier die erste wieder übereinstimmende Zeile, also:

```
.
.
.
****** Datei1
Unterschiede
1. wieder übereinstimmende Zeile
****** Datei 2
Unterschiede
1. wieder übereinstimmende Zeile
.
.
.
```

Wenn die Anzahl von Zeilen im Textpuffer kleiner ist, als die Anzahl von aufeinander folgenden, sich unterscheidenden Zeilen, hält FC mit dem Vergleich an, zeigt die Meldung

```
resynch failed. Files are too different.
```

(Keine Übereinstimmung. Dateien sind zu verschieden) sowie daran anschließend den Inhalt des Textpuffers und kehrt schließlich auf die MS–DOS–Befehlsebene zurück.

9.6 Umleitung der FC–Ausgabe in eine Datei

Normalerweise gibt FC sein Vergleichsprotokoll auf dem Bildschirm aus. Mit Hilfe des Umleitungssymbols von MS–DOS (>) kann die Ausgabe jedoch leicht in eine Datei oder auf den Drucker umgeleitet werden (s.a. Kapitel 5.5.1).

Um z.B. das Protokoll des Vergleiches der beiden Dateien TEXT1.TXT und TEXT2.TXT für eine spätere Auswertung in die Datei DIFFER.TXT auszugeben, kann folgender Aufruf von FC eingegeben werden:

```
fc text1.txt text2.txt > differ.txt
```

Bei einer direkten Ausgabe auf Drucker lautet der Aufruf:

```
fc text1.txt text2.txt > prn
```

9.7 Beispiele

Angenommen, es extistierten die beiden folgenden Dateien:

TEXT1.TXT

```
Er ist da . . .
Der neue X-1000-Automatic-Bleistiftanspitzer
von Spitzer-Bürohilfsmittel,
dem Weltführer der Büro-Anspitztechnik

Dieses Modell ist kompakt und spitzt
Bleistifte effizienter als je zuvor.

Unser Motto:
"Auch Sie sollten Spitze sein"

Hans Spitzer, Direktor
```

TEXT2.TXT

```
Er ist da . . .
Der neue X-2000-Automatic-Stift- und -Bleistiftanspitzer
von Spitzer-Bürohilfsmittel,
dem Weltführer der Büro-Anspitztechnik

Dieses Modell ist kompakt und spitzt Stifte und
Bleistifte effizienter als je zuvor.

Unser Motto:
"Auch Sie sollten Spitze sein"

Hans Spitzer, Direktor
```

Um die beiden Dateien miteinander zu vergleichen und die Unterschiede auf dem Bildschirm anzuzeigen, geben Sie ein:

```
fc text1.txt text2.txt
```

Dabei gelten folgende Voreinstellungen des FC–Programmes:

ASCII–Vergleich mit einer Textpufferlänge von 100 Zeilen. Tabulatorzeichen werden bis zur nächsten 8–Spaltenposition mit Leerstellen expandiert. Es wird zwischen Groß– und Kleinschreibung unterschieden, Zeilennummern werden nicht angezeigt und Mehrfachleerstellen werden nicht komprimiert.

Auf dem Bildschirm erscheint dann das folgende Protokoll:

```
***** text1.txt
Er ist da . . .
Der neue X-1000-Automatic-Bleistiftanspitzer
von Spitzer-Bürohilfsmittel,

***** text2.txt
Er ist da . . .
Der neue X-2000-Automatic-Stift- und -Bleistiftanspitzer
von Spitzer-Bürohilfsmittel,
*****

***** text1.txt

Dieses Modell ist kompakt und spitzt
Bleistifte effizienter als je zuvor.
***** text2.txt

Dieses Modell ist kompakt und spitzt Stifte und
Bleistifte effizienter als je zuvor.
*****
```

Soll dagegen das Protokoll auf dem Drucker ausgedruckt werden und sollen mindestens 4 aufeinanderfolgende Zeilen für eine gültige Übereinstimmung in beiden Dateien gleich sein, so müssen Sie eingeben:

```
fc /4 text1.txt text2.txt >prn
```

Sie erhalten dann folgendes Druckprotokoll:

```
***** text1.txt
Er ist da . . .
Der neue X-1000-Automatic-Bleistiftanspitzer
von Spitzer-Bürohilfsmittel,
dem Weltführer der Büro-Anspitztechnik
```

```
Dieses Modell ist kompakt und spitzt
Bleistifte effizienter als je zuvor.
***** text2.txt
Er ist da . . .
Der neue X-2000-Automatic-Stift- und -Bleistiftanspitzer
von Spitzer-Bürohilfsmittel,
dem Weltführer der Büro-Anspitztechnik

Dieses Modell ist kompakt und spitzt Stifte und
Bleistifte effizienter als je zuvor.
*****
```

Wenn Sie dagegen die beiden o.a. Textdateien binär vergleichen und das Protokoll auf dem Bildschirm anzeigen wollen, so geben Sie ein:

```
fc /b text1.txt text2/txt
```

und Sie erhalten:

```
0000001A: 31 32
00000029: 42 53
0000002A: 6C 74
0000002B: 65 69
0000002C: 69 66
0000002D: 73 74
0000002E: 74 2D
0000002F: 69 20
00000030: 66 75
00000031: 74 6E
.
.
.
00000111: 44 65
00000113: 72 6E
00000114: 65 22
00000115: 6B 0D
00000116: 74 0A
00000117: 6F 0D
00000119: 71 0A
0000011A: 0A 61
0000011B: 1A 6E
fc: text2.txt longer than text1.txt
```

10. LINK – Ein Bindeprogramm für Objektdateien

In diesem Kapitel lernen Sie:

- Wie mit LINK ausführbare Programmdateien erzeugt werden können;
- wie die Befehlsparameter beim Aufruf von LINK verwendet werden können;
- wie LINK arbeitet.

10.1 Einführung

Der Microsoft–8086–Object–Linker LINK erzeugt ausführbare Programmdateien, indem er Objekt–Dateien zusammenbindet, die mit dem Microsoft-Macro–Assembler MASM oder mit Hochsprachen–Compilern wie z.B. dem C–Compiler oder dem GW–BASIC–Compiler Quickbasic von Microsoft erstellt wurden. Das gebundene, ausführbare Programm wird von LINK in eine ausführbare Datei (.EXE–Datei) kopiert. Dieses Programm kann dann unter MS–DOS einfach durch Eingabe des Dateinamens (die Standard–Erweiterung .EXE braucht nicht angegeben zu werden) in der MS–DOS–Befehlszeile aufgerufen und ausgeführt werden.

LINK benötigt eine oder mehrere Objekt–Dateien, die zusammen mit den erforderlichen Routinen aus der LINK–Bibliothek zum fertigen Programm zusammengebunden werden. Dabei werden Code und Daten in den Obektdateien miteinander kombiniert und die angegebenen Bibliotheken durchsucht, um externe Referenzen den Routinen und Variablen zuzuweisen. Anschließend wird ein verschieb– und ausführbares Abbild des Programms zusammen mit den Verschiebungs–Informationen in die Programmdatei kopiert. Mit Hilfe der Verschiebungsinformationen kann MS–DOS das ausführbare Programmabbild in jeden passenden Speicherbereich laden und dort starten. LINK kann Programme mit einem Umfang von bis zu 1 MBytes Programmcode und –daten verarbeiten.

In diesem Kapitel lernen Sie, wie Sie den Binder LINK zur Erzeugung ausführbarer Programmdateien benutzen können. Alle beim Aufruf von LINK spezifizierbaren Optionen (Programm–Parameter) zur Steuerung des Bin-

de–Ablaufes werden detailliert beschrieben und die Arbeitsweise von LINK wird erläutert.

10.2 LINK aufrufen und benutzen

In diesem Abschnitt werden drei Methoden für den Aufruf und die Benutzung des Binders LINK zur Erzeugung ausführbarer Programme beschrieben. Dabei wird auf eine Reihe von Programmanforderungen entweder durch Tastatureingaben, durch Eingabe einer MS–DOS–Befehlszeile oder durch Verwendung einer Parameterdatei geantwortet. Alle drei Methoden können auch gemischt angewendet werden.

Nach dem Start verarbeitet LINK entweder die beim Aufruf angegebenen Objektdateien oder fordert die Eingabe von Dateinamen an. LINK kann zu jedem Zeitpunkt durch Drücken von Ctrl–C oder Ctrl–Break abgebrochen werden.

10.2.1 Parameteranforderung

Wenn Sie auf der MS–DOS–Befehlsebene nur

```
link
```

eingeben, fordert LINK nach und nach die benötigten Informationen wie Objektdateien und Optionen an. Folgen Sie dabei diesen Schritten:

1. Geben Sie zunächst den folgenden Befehl ein und drücken Sie zum Abschluß die Eingabetaste:

   ```
   link
   ```

   LINK fordert dann mit der folgenden Meldung die zu bindenden Objekt-Dateien an:

   ```
   Object Modules [.OBJ]:
   ```

2. Geben Sie jetzt den oder die Namen der zu bindenden Objektdatei(en) ein. Wenn keine Namenserweiterung benutzt wird, nimmt LINK die Erweiterung .OBJ an. Werden mehrere Namen eingegeben, so müssen

diese mit dem Pluszeichen (+) voneinander getrennt werden. Passen die Namen nicht in eine Befehlszeile, so muß nach dem letzten Namen in der Zeile ein Pluszeichen (+) eingegeben und dann die Eingabetaste gedrückt werden. LINK fordert dann weitere Objektdateinamen an. Am Ende der Namensliste wird die Eingabetaste gedrückt. LINK fordert dann mit folgender Meldung den Namen der ausführbaren Programmdatei an:

```
Run File [Datei.EXE]:
```

3. Dabei wird für *Datei* der erste in der Objektdatei–Namensliste angegebene Namen gesetzt. Geben Sie jetzt den Namen, den das fertige Programm erhalten soll ein, und drücken Sie die Eingabetaste. Wenn Sie keine Namenserweiterung angeben, nimmt LINK .EXE an. Wenn Sie den angebotenen Dateinamen akzeptieren, drücken Sie nur die Eingabetaste. Die Programmdatei erhält dann denselben Namen, wie die erste unter 1. angegebene Objektdatei und die Namenserweiterung .EXE. Nachdem Sie also hier die Eingabetaste gedrückt haben, fordert LINK mit folgender Meldung den Namen der List–Datei, in die für jedes Segment in den Objekt–Moduln ein Adreßeintrag gemacht wird:

```
List File [NUL.MAP]:
```

4. Geben Sie hier den Namen der Listdatei, die auch als LINK–Programmabbild bezeichnet wird, ein und drücken Sie die Eingabetaste. Wenn Sie keine Namenserweiterung angeben, benutzt LINK die Erweiterung .MAP. Wenn Sie überhaupt keine List–Datei wollen, brauchen Sie nur die Eingabetaste zu drücken. Nach dem Drücken der Eingabetaste fordert LINK mit der folgenden Meldung die Eingabe der Bibliotheksdatei(en):

```
Libraries [.LIB]:
```

5. Geben Sie jetzt den/die Namen der Bibliotheksdatei(en) ein, aus der/denen LINK die Routinen oder Variablen kopiert, auf die Sie in Ihrem Programm Bezug genommen haben, die aber dort nicht enthalten sind. Werden mehrere Namen eingegeben, so müssen diese mit dem Pluszeichen (+) voneinander getrennt werden. Passen die Namen nicht in eine Befehlszeile, so muß nach dem letzten Namen in der Zeile ein Pluszeichen (+) eingegeben und dann die Eingabetaste gedrückt werden. LINK fordert dann weitere Bibliotheksdateinamen an. Nach der Eingabe des letzten Namens drücken Sie die Eingabetaste. Wenn keine Bibliotheksdatei benutzt werden soll, brauchen Sie hier nur die Eingabetaste zu drücken.

LINK erzeugt jetzt die ausführbare Programmdatei.

Befindet sich eine der oben erwähnten Dateien nicht im aktuellen Verzeichnis der Disk im aktuellen Laufwerk, so müssen Sie den vollständigen Pfadnamen für die bezogene Datei angeben. Die LINK–Optionen (Parameter für Befehlserweiterungen (s. Kapitel 10.3)) können bei jeder Anforderung nach dem jeweiligen Dateinamen angegeben werden. Wenn der Binder eine angegebene Objektdatei nicht findet, meldet er dies und fordert, falls nötig, zum Diskettenwechsel auf.

Bei jeder Anforderung können Sie den Rest der benötigten Dateinamen und Optionen mit angeben, indem Sie das im nächsten Abschnitt beschriebene Befehlszeilenformat von LINK verwenden. So können Sie z.B. für alle verbleibenden Anforderungen die Voreinstellung wählen, indem Sie als Abschlußzeichen bei einer Anforderung ein Semikolon (;) setzen. Mehrere Dateien können durch Angabe von Kommata (,) spezifiziert werden. Wenn Sie bei der Objektdatei–Anforderung die Eingabe mit einem Semikolon beenden, sollten Sie mindestens einen Objektdateinamen vorher eingegeben haben. Sobald der Binder ein Semikolon bei der Parametereingabe findet, wählt er für alle verbleibenden Anforderungen die Voreinstellung und beginnt den Bindevorgang ohne weitere Anforderungen.

Beispiel: Beim folgenden Beispiel werden die Objekt–Module MODA.OBJ, MODB.OBJ, MODC.OBJ und START.OBJ zusammengebunden. Dabei wird die Bibliotheksdatei MATH.LIB im Unterverzeichnis \LIB auf der Disk in Laufwerk B verwendet. Die ausführbare Programmdatei soll den Namen MODA.EXE erhalten und es soll eine Listdatei mit Namen ABC.MAP erstellt werden. Der /PAUSE–Schalter in der zweiten Anforderungszeile für die Objektdateinamen veranlaßt LINK, vor der Erstellung der ausführbaren Progammdatei zu warten, bis die Diskette gewechselt wurde (s.a. Kapitel 10.3.1):

```
link

Object Modules [.OBJ]: moda+modb+
Object Modules [.OBJ]: modc+start/pause
Run File [moda.EXE]:
List File [NUL.MAP]: abc
Libraries [.LIB]: b:\lib.math
```

10.2.2 Der vollständige Aufruf

Sie können ein ausführbares Programm erzeugen, indem Sie das Befehlswort LINK gefolgt von den Namen der zu bindenden Dateien sowie den benötigten Parametern eingeben. Das allgemeine Format des Aufrufs von LINK in einer Befehlszeile lautet:

[*L*:][*BPfad*]**LINK**[*Laufw*:][*Pfad*]*ODatei*[*.Erw*]
[+[*Laufw*:][*Pfad*][*Odatei*[*.Erw*]]. . .
[,[*Laufw*:][*Pfad*][*ADatei*][*.Erw*]
[,[*Laufw*:][*Pfad*][*LDatei*][*.Erw*]
[,[*Laufw*:][*Pfad*][*BDatei*][*.Erw*]
[+[*Laufw*:][*Pfad*][*Bdatei*[*.Erw*]]. . .]]][*Param*][;]

Dabei bedeuten:

L:BPfad Laufwerk und Suchpfad für die LINK–Programmdatei.

Laufw:Pfad Odatei.Erw Die vollständige(n) Angabe(n) für die Objektdatei(en), die gebunden werden soll(en). Wird ***Laufw:*** weggelassen, so nimmt LINK das aktuelle Laufwerk an. Wird ***Pfad*** weggelassen, so nimmt LINK das aktuelle Verzeichnis der Disk im angegebenen Laufwerk an. Wird ***.Erw*** weggelassen, so nimmt LINK .OBJ an.

Laufw:Pfad Adatei.Erw Die vollständige Angabe für die ausführbare Programmdatei, die gebunden werden soll. Wird ***Laufw:*** weggelassen, so nimmt LINK das aktuelle Laufwerk an. Wird ***Pfad*** weggelassen, so nimmt LINK das aktuelle Verzeichnis der Disk im angegebenen Laufwerk an. Wird ***Adatei*** weggelassen, so nimmt LINK den Namen der ersten Objektdatei (s.o.) an. Wird ***.Erw*** weggelassen, so nimmt LINK .EXE an.

Laufw:Pfad LDatei.Erw Die vollständige Angabe für die List–Datei, in die für jedes Segment in den Objekt–Moduln ein Adreßeintrag gemacht wird. Wird ***Laufw:*** weggelassen, so nimmt LINK das aktuelle Laufwerk an. Wird ***Pfad*** weggelassen, so nimmt LINK das aktuelle Verzeichnis der Disk im angegebenen Laufwerk an. Wird ***LDatei*** weggelassen, so nimmt LINK den Namen NUL an, erstellt also keine Listdatei. Wird ***.Erw*** weggelassen, so nimmt LINK .MAP an.

Laufw:Pfad Bdatei.Erw Die vollständige(n) Angabe(n) für die Bibliotheksdatei(en), aus der/denen LINK die Routinen oder Variablen kopiert, auf die Sie in Ihrem Programm Bezug genommen haben, die aber dort nicht enthalten sind. Wird ***Laufw:*** weggelassen, so nimmt LINK das aktuelle

Laufwerk an. Wird ***Pfad*** weggelassen, so nimmt LINK das aktuelle Verzeichnis der Disk im angegebenen Laufwerk an. Wird die ganze Dateiangabe weggelassen, so wird keine Bibliotheksdatei zum Binden benutzt. Wird ***.Erw*** weggelassen, so nimmt LINK .LIB an.

Param Eine Liste von Parametern, mit denen der Bindevorgang gesteuert werden kann. Diese Parameter erweitern die Möglichkeiten des LINK–Programms erheblich und werden deshalb gesondert im Kapitel 10.3 behandelt.

Die Dateiangaben müssen für jede angegebene Datei vollständig sein. LINK geht also **nicht** davon aus, daß Laufwerk und Suchpfad der ersten angegebenen Objektdatei auch für alle anderen Dateien gelten.

Wird nach der Liste der Objektdateien ein Komma (,) angegeben, verwendet LINK den voreingestellten Namen für die ausführbare Programmdatei und unterdrückt List– und Bibliotheksdateien. Ein Semikolon (;) an beliebiger Stelle in der Befehlszeile schließt diese ab.

Werden nicht alle Dateinamen in der Befehlszeile angegeben und wird diese auch nicht mit einem Semikolon (;) abgeschlossen, so fordert LINK mit den in Kapitel 10.2.1 beschriebenen Meldungen weitere Dateiangaben an.

Beachten Sie: Wenn Sie Objektmodule binden wollen, die mit einem Hochsprachen–Compiler erzeugt wurden, der Overlays (Überlagerungs-Programmodule) unterstützt, müssen Sie diese Module in Klammern setzen. Da der Microsoft–Makroassembler MASM keine Overlay–Verwaltung kennt. können Sie Overlay–Module nur für Objektdateien angeben, die mit der Laufzeit–Bibliothek eines Hochsprachen–Compilers gebunden wurden, der Overlays unterstützt. So können Sie Overlays mit Moduln verwenden, die mit den Microsoft–Compilern FORTRAN 3.2 ff, Pascal 3.2 ff und C 3.2 ff übersetzt wurden. In den jeweiligen Compiler–Handbüchern finden Sie weitere Informationen für die Verwendung von Overlays.

Beispiele

Im ersten Beispiel wird die Objektdatei PROG.OBJ zur ausführbaren Datei PROG.EXE gebunden. Dabei wird die Bibliothek MATH.LIB für Daten– und Variablenreferenzen benutzt, die im Programm verwendet wurden. Es wird außerdem die Listdatei ABB.MAP erzeugt, in der eine Liste der Programmsegmente und –gruppen angelegt wird. Alle Dateien befinden sich im aktuellen Verzeichnis der Disk im aktuellen Laufwerk oder werden dort angelegt:

```
link prog.obj,prog.exe,abb.map,math.lib
```

Diese Befehlszeile kann aber auch wesentlich kürzer geschrieben werden und bedeutet dennoch dasselbe:

```
link prog,,,math
```

Im zweiten Beispiel werden die beiden Objektdateien PROG.OBJ und START.OBJ im aktuellen Verzeichnis der Disk im aktuellen Laufwerk zu einer ausführbaren Programmdatei PROG.EXE gebunden, die im aktuellen Verzeichnis der Disk in Laufwerk B erzeugt werden soll. Die List–Datei ABB.MAP soll im Unterverzeichnis \MAP auf der Disk im aktuellen Laufwerk erzeugt werden und es soll keine Bibliotheksdatei zum Binden verwendet werden:

```
link start+prog,b:prog,\map\abb;
```

Im letzten Beispiel werden die vier Objektmoduln MODA.OBJ, MODB.OBJ, MODC.OBJ und START.OBJ zur ausführbaren Programmdatei moda.exe gebunden. Alle diese Dateien befinden sich im aktuellen Verzeichnis der Disk im aktuellen Laufwerk. Zum Binden wird die Bibliotheksdatei MATH.LIB im \LIB–Unterverzeichnis der Disk im Laufwerk B verwendet. Die Listdatei ABC.MAP wird wieder im aktuellen Verzeichnis der Disk im aktuellen Laufwerk angelegt. Die Option /PAUSE wartet mit der Erzeugung der ausführbaren Programmdatei, bis Disketten gewechselt wurden (s.a. Kapitel 10.3.1):

```
link moda+modb+modc+start/pause,,abc,b:\lib\math
```

10.2.3 Angabe einer Parameterdatei

Das Binden komplexer Programme efordert häufig eine recht umfangreiche Angabe von Parametern für das LINK–Programm. In solchen Fällen, insbesondere, wenn das komplexe Programm häufig übersetzt und wieder gebunden werden soll, ist es möglich, alle Dateinamen und Parameter für das LINK–Programm in einer Parameterdatei zusammenzustellen und LINK dann mit dem Namen dieser Parameterdatei aufzurufen. Dem Namen muß dann das At–Symbol (@) vorangestellt werden, also:

```
link @pardatei
```

Die Parameterdatei kann auch bei jeder Parameteranforderung von LINK angegeben werden oder auch an jeder Position in der Befehlszeile. LINK liest die Parameterdatei genauso, als kämen die Eingaben von der Tastatur. Jede

Kombination Wagenrücklauf/Zeilenvorschub in der Parameterdatei wird so behandelt, wie das Drücken der Eingabetaste auf der Tastatur bei der Beantwortung einer Parameteranforderung oder wie ein Komma (,) in der Befehlszeile. Die Parameterdatei hat also die allgemeine Form:

Objektdateien
[***Programmdatei***]
[***Listdatei***]
[***Bibliotheksdatei***]

Befindet sich die Parameterdatei in einem anderen Verzeichnis als dem aktuellen oder auf einer Disk in einem anderen Laufwerk, so müssen Pfad– bzw. Laufwerksname mit angegeben werden.

Die Parameterdatei kann analog zu einer Stapelverarbeitungsdatei z.B. mit dem Zeileneditor EDLIN (s. Kapitel 8) erzeugt werden. Es können alle Parameter, die bereits bei LINK–Anforderungen eingegeben oder als Teile der Befehlsdatei bereits verarbeitet wurden, weggelassen werden. Jede Dateinamengruppe muß in einer eigenen Zeile angegeben werden. Passen nicht alle zu einer Gruppe gehörigen Namen in eine Zeile, so kann als letztes Zeichen in der Zeile das Pluszeichen (+) angegeben werden und dann weitere Namen in der Folgezeile. Wird eine Gruppe weggelassen, so muß an der Stelle in der Parameterdatei eine Leerzeile gelassen werden. LINK–Optionen können auf jeder Zeile der Parameterdatei angegeben werden. Ebenso ein Semikolon (;). Findet LINK in der Parameterdatei ein Semikolon, verwendet es automatisch für alle bis dahin noch nicht angegebenen Dateinamen die voreingestellten Werte und ignoriert den Rest der Parameterdatei.

Wenn Sie ein Programm mit Hilfe einer Parameterdatei binden, zeigt der Binder jede Antwort aus der Parameterdatei wie eine beantwortete Anforderung an. Enthält die Parameterdatei keine Angaben für benötigte Dateien, so werden diese von LINK angefordert und können über die Tastatur eingegeben werden.

Beachten Sie: Eine Parameterdatei sollte immer mit einem Semikolon (;) oder einer Wagenrücklauf/Zeilenvorschub–Kombination abgeschlossen werden. Fehlt eines dieser beiden Abschlußzeichen, so zeigt LINK die letzte Zeile der Parameterdatei auf dem Bildschirm an und wartet, bis Sie die Eingabetaste gedrückt haben.

Beispiel

Die folgende Parameterdatei weist den Binder an, die vier Objektdateien MODA, MODB, MODC und START zu binden. Vor der Erstellung der ausführbaren Programmdatei MODA.EXE soll auf Diskettenwechsel gewartet werden. Es soll eine List–Datei ABC.MAP erzeugt werden und die Bibliotheksdatei MATH.LIB im Unterverzeichnis \LIB auf der Disk im Laufwerk B benutzt werden:

```
moda+modb+modc+start /pause
abc
b:\lib\math
```

In der folgenden Prozedur werden alle drei Möglichkeiten der Dateinamen–Spezifizierung gezeigt. Angenommen Sie haben eine Parameterdatei namens PARAM erzeugt, die nur eine Zeile enthält, nämlich:

```
lib1+lib2+lib3+lib4
```

Jetzt starten Sie LINK mit folgender Teil–Befehlszeile:

```
link objekt1+objekt2
```

LINK nimmt die beiden Dateien OBJEKT1 und OBJEKT2 als Objektdateien und fordert weitere Dateien an:

```
Run File [objekt1.EXE]: exec
List File [NUL.MAP]:
Libraries [.LIB]: @param
```

Sie haben den Namen EXEC eingegeben, womit LINK die ausführbare Programmdatei EXEC.EXE erzeugt. Bei der Listdatei–Anforderung haben Sie die Eingabetaste gedrückt, weil keine Listdatei erstellt werden soll. Bei der Anforderung der Bibliotheksdatei haben Sie den Namen Ihrer Parameterdatei PARAM eingegeben, in der die vier Bibliotheksdateien, die zum Binden verwendet werden sollen, angegeben sind.

10.2.4 Angabe von Suchpfaden für Bibliotheken

Sie können LINK veranlassen, nach Laufwerken oder Verzeichnissen, die Sie zusammen mit den Bibliotheksnamen in der Befehlszeile angegeben haben,

zu suchen, indem Sie entweder einen oder mehrere Suchpfade bei den Dateinamen eingeben oder indem Sie diese Suchpfade der Umfeldvariablen LIB zuweisen (s. SET–Befehl im Kapitel 6.3.1). Im letzteren Fall müssen Sie die Suchpfade durch Semikolon (;) trennen.

Ein Suchpfad ist der Pfad eines Verzeichnisses oder Laufwerkes. Suchpfade werden zusammen mit den Dateinamen in der Befehlszeile von LINK oder in Beantwortung einer Anforderung von LINK eingegeben.

Wenn Sie ein Laufwerk oder einen Suchpfad bei einem Bibliotheksdateinamen in der LINK–Befehlszeile angeben, sucht der Binder nur dort nach der Datei. Wird **kein** Laufwerk oder Suchpfad angegeben, sucht LINK nach der Datei in dieser Reihenfolge:

1. Zunächst wird das aktuelle Verzeichnis der Disk im aktuellen Laufwerk durchsucht.
2. Wird die Bibliothek dort nicht gefunden und sind weitere Suchpfade in der Befehlszeile angegeben, durchsucht der Binder diese in der angegebenen Reihenfolge.
3. Wird die Bibliothek auch dort nicht gefunden und wurde der LIB–Variablen in der Umfeldzeichentabelle von MS–DOS ein Suchpfad zugewiesen, so wird dann dieser durchsucht.
4. Wenn die Bibliothek noch immer nicht gefunden wird, zeigt LINK eine Fehlermeldung an.

Beispiele

Im ersten Beispiel sucht LINK nur im Verzeichnis \ALTLIB auf der Disk im Laufwerk A nach der Bibliotheksdatei MATH.LIB. Um die Bibliotheksdatei COMMON.LIB zu finden, durchsucht LINK das aktuelle Verzeichnis der Disk im aktuellen Laufwerk, das aktuelle Verzeichnis der Disk in Laufwerk B und schließlich das Verzeichnis \lib der Disk in Laufwerk D:

```
link prog,,prog,a:\altlib\math+common+b:+d:\lib\
```

Im zweiten Beispiel durchsucht LINK das aktuelle Verzeichnis der Disk im aktuellen Laufwerk, das \LIB–Verzeichnis der Disk in Laufwerk C und das \SYSTEM\LIB–Verzeichnis der Disk in Laufwerk U nach den Bibliotheksdateien MATH.LIB und COMMON.LIB:

```
set lib=c:\lib;u:\system\lib
link prog,,prog,math+common
```

10.2.5 Die List–Datei (MAP–Datei)

Die List–Datei wird auch als Programmabbild–Datei bezeichnet, weil in ihr ein Abbild des gebundenen Programms in Form von einzelnen Einträgen abgelegt wird. Jeder Eintrag enthält den Namen, die Ladeadresse und die Länge für ein Programmsegment. Dieselben Einträge werden auch für jede Gruppe im Programm gemacht. Außerdem werden hier die Programmstartadresse sowie jegliche Fehlermeldungen, die beim Binden abgesetzt wurden, hinterlegt. Wird außerdem die /MAP–Option in der LINK–Befehlszeile angegeben, so werden in der List–Datei die Namen und Ladeadressen aller Public–Symbole (s. dazu *MASM Referenz Manual*) abgelegt.

Die Segmentinformationen werden in der folgenden Form (Beispiel) in der List–Datei abgelegt:

```
Start     Stop      Length    Name    Class

00000H    0172CH    0172DH    TEXT    CODE
01730H    01E19H    006EAH    DATA    DATA
```

Die Spalten Start und Stop enthalten die 20–Bit–Adressen in hexadezimaler Codierung des ersten und letzten Bytes eines jeden Segmentes. Diese Angaben sind die relativen Adressen bezogen auf den Anfang des Lademoduls bei der Adresse 00000H. Das Betriebssystem wählt seine eigene Startadresse, sobald das Programm tatsächlich geladen ist. Die Length–Spalte gibt die Segmentlänge in Bytes an. In der Name–Spalte steht der Segmentname und die Class–Spalte enthält den Klassen–Namen des Segmentes.

Die Gruppen–Informationen werden folgendermaßen in der List–Datei abgelegt:

```
Origin          Group

0000:0          IGROUP
0173:0          DGROUP
```

Bei diesem Beispiel ist IGROUP der Name der Code– (Befehls–)Gruppe und DGROUP der Name der Daten–Gruppe.

Am Ende der List–Datei wird die Adresse für den Programmeinsprung angegeben.

Wurde die Option /MAP in der LINK–Befehlszeile angegeben, wird zusätzlich eine Liste der Namen der Public–Symbole (Allgemeine Symbole) in die List–Datei ausgegeben. Alle Symbole werden dabei zweimal angegeben. Einmal in alphabetischer Namensordnung und einmal geordnet nach aufsteigenden Ladeadresse. Die folgende Liste ist ein Beispiel:

```
Adress                  Publics by Name

0000:1567               BRK
0000:1696               CHMOD
0000:01DB               CHKSTK
0000:131C               CLEARERR
0173:0035               FAC

Adress                  Publics by Value

0000:01DB               CHKSTK
0000:131C               CLEARERR
0000:1567               BRK
0000:1696               CHMOD
0173:0035               FAC
```

Die Adressen der Public–Symbole werden in der Form Segment:Offset angegeben und zeigen den Platz des Symbols relativ zum Anfang des Lademoduls bei der Adresse 0000:0000.

Wurden die Optionen /HIGH und /DSALLOCATE (s. dort in Kapitel 10.3.1) benutzt und Programm–Code und –Daten belegen zusammen nicht mehr als 64 KBytes Speicher, kann die List–Datei ungewöhnlich große Segmentadressen enthalten. Solche Adressen weisen ein Symbol aus, dessen Platz unterhalb der Adresse liegt, bei der der Pogrammcode und die –daten beginnen. Z.B. liegt das Symbol FELD unterhalb der Startadresse, wenn folgender Eintrag dafür ausgewiesen wird:

```
FFF0:0A20               FELD
```

Die 20–Bit–Adresse von FELD ist in diesem Fall 00920H.

10.2.6 Die temporäre Arbeitsdatei VM.TMP

Normalerweise benutzt LINK den verfügbaren Speicher für das Zusammenbinden der Objektdateien. Reicht dieser Speicher bei sehr großen Programmen nicht aus, so wird im aktuellen Verzeichnis der Disk im aktuellen Laufwerk die Arbeitsdatei VM.TMP eingerichtet und die Meldung

```
VM.TMP has been created
Do not change Diskette in Drive x:
```

(VM.TMP wurde angelegt. Diskette in Laufwerk *x* nicht wechseln) angezeigt. In diesem Fall darf die Diskette im angegebenen Laufwerk nicht gewechselt werden, bis LINK beendet ist. Außerdem kann hier auch die /PAUSE–Option **nicht** benutzt werden. Nach dem Erzeugen der ausführbaren Programmdatei wird die temporäre Datei VM.TMP wieder gelöscht.

Beachten Sie: Benutzen Sie den Namen VM.TMP nicht für eigene Dateien, weil LINK diese Datei ggf. überschreiben und damit ihren Inhalt zerstören würde.

10.3 Befehlserweiterungen mit den LINK–Optionen

Mit den LINK–Optionen kann der Bindevorgang gesteuert werden. Jede Option beginnt mit einem Schrägstrich (/), gefolgt von einem Schlüsselwort. Folgende Optionen können an beliebiger Stelle in der LINK–Befehlszeile benutzt werden:

Name der Option	Bedeutung
/CPARMAXALLOC	setzt maximal zu reservierenden Programmspeicher;
/DOSSEG	benutzt die MS–DOS–Segmentordnung;
/DSALLOCATE	reserviert Speicher für eine Datengruppe;
/EXEPACK	erzeugt eine ausführbare Programmdatei in optimierter Größe;
/HELP	zeigt die Liste der erlaubten Optionen;
/HIGH	setzt die höchst–mögliche Startadresse;
/LINENUMBERS	trägt die Startadresse jeder Quellprogrammzeile in die List–Datei ein;
/MAP	fügt in die List–Datei eine Liste der Public-Symbole ein;
/NODEFAULTLIBRARYSEARCH	ignoriert Bibliotheksnamen in Objektdateien;
/NOGROUPASSOCIATION	löscht Gruppenzugehörigkeiten;
/NOIGNORECASE	unterscheidet zwischen Groß– und Kleinbuchstaben in Namen;
/OVERLAYINTERRUPT	setzt einen Interrupt für den Overlay–Lader;
/PAUSE	unterbricht den Bindevorgang zum Diskettenwechsel;
/SEGMENTS	setzt maximale Segmentzahl;
/STACK	setzt die Programm–Stapelspeichergröße.

Die Namen der Optionen dürfen auch abgekürzt werden, solange sich der abgekürzte Name vom Namen anderer Optionen unterscheidet. Die maximalen Abkürzungen sind bei den einzelnen Beschreibungen beim Format angegeben.

Viele der LINK–Optionen setzen Werte in den MS–DOS–Progammkopf für .EXE–Dateien. Solche Optionen können besser verstanden werden, wenn Sie die Organisation des Programmkopfes kennen (s. Handbuch *MS–DOS Programmer's Reference*).

10.3.1 Die LINK–Optionen im einzelnen

In den folgenden Abschnitten wird jede der LINK–Optionen ausführlich beschrieben. Das Format und die Syntax der Beschreibungen entsprich dabei denen bei der Beschreibung der MS–DOS–Befehle (s.a. Kapitel 6.2). Jede einzelne Befehlsbeschreibung beginnt dabei auf einer neuen Seite.

/CPARMAXALLOC

Format: **/C[PARMAXALLOC]:*Zahl***

Bedeutung: Setzt die maximale Zahl von 16–Byte–Speicherblöcken, die ein Programm braucht, wenn es in den Speicher geladen wird. MS–DOS benutzt diese Zahl, wenn es vor dem Laden eines Programmes diesem Speicherplatz zuweist.

Zahl Eine ganze Zahl zwischen 1 und 65535 in dezimaler, oktaler oder hexadezimaler Codierung. Oktale Zahlen müssen mit einer 0 und hexadezimale mit einer 0 gefolgt vom Kleinbuchstaben **x** (z.B. 0x2B) beginnen.

Beispiele: Es werden 15 16–Byte–Blöcke reserviert:

```
link prog/c:15,,;
```

Hier werden 255 16–Byte–Blöcke reserviert:

```
link moda+modb,run/cparm:0xff,ab;
```

Schließlich noch ein Beispiel für oktale Schreibweise (20 Blöcke):

```
link start+prog,/c:030,;
```

Anmerkungen:

- LINK setzt normalerweise die maximale Zahl der Speicherblöcke auf 65535. Da dieser Wert aber den gesamten adressierbaren Speicher repräsentiert, ignoriert MS–DOS die voreingestellten Werte und weist den größten zusammenhängenden Speicherbereich, den es finden kann, zu. Wenn dagegen die /CPARMAXALLOC–Option benutzt wird, weist MS–DOS nicht mehr als den dabei angegebenen Speicher zu, so daß jeder weitere Speicherbreich frei für andere Programme bleibt.

- Wird für ***Zahl*** ein kleinerer Wert als die minimal vom Programm benötigte Anzahl von 16–Byte–Blöcken

angegeben, ignoriert LINK diesen Wert und setzt den maximalen Wert auf den minimal benötigten. Die kleinste von einem Programm benötigte Anzahl von 16–Byte–Blöcken ist nie kleiner als die Anzahl von Code– und Daten–Blöcken in diesem Programm.

- Diese Option kann maximal auf /C abgekürzt werden.

/DOSSEG

Format: **/DO[SSEG]**

Bedeutung: Veranlaßt LINK, alle Segmente in der ausführbaren Programmdatei entsprechend den MS–DOS–Segmentordnungs–Konventionen zu ordnen. Diese Konventionen halten sich an folgende Regeln:

- Alle Segmente mit dem Klassennamen CODE werden an den Anfang der ausführbaren Programmdatei plaziert.
- Alle Segmente, die **nicht** zur Gruppe DGROUP gehören, werden nach den CODE–Segmenten angeordnet.
- Alle Segmente, die zu DGROUP gehören, werden ans Ende der ausführbaren Progammdatei plaziert.

Beispiel: Die folgende LINK–Befehlszeile veranlaßt den Binder, die ausführbare Programmdatei PROG.EXE zu erzeugen, in der alle Segmente entsprechend den MS–DOS–Segmentordnungs–Konventionen angeordnet werden. Die Segmente in den Objektdateien START.OBJ und TEST-.OBJ sowie alle aus den beiden Bibliotheksdateien MATH.LIB und COMMON.LIB werden nach denselben Konventionen, wie oben beschrieben, angeordnet:

```
link start+test/dosseg,,,math+common
```

Anmerkungen:

- Wenn Sie die /DOSSEG–Option nicht verwenden wollen, finden Sie in Kapitel 10.4.3 die Beschreibung der normalen Segmentanordnung.
- Diese Option kann maximal auf /DO abgekürzt werden.

/DSALLOCATE

Format: **/D[SALLOCATE]**

Bedeutung: Weist den Binder an, bei der Zuweisung von Adressen zu Programmelementen, die zur Gruppe DGROUP gehören, umgekehrt als normal zu verfahren. Normalerweise weist LINK dem ersten Byte einer Gruppe den Adreßoffset 0000H zu. Mit der Option /DSALLOCATE weist LINK dem letzten Byte einer Grupe den Offset FFFFH zu. Das führt dann dazu, daß Daten an die höchst mögliche Adresse des Speichersegmentes geladen werden, das DGROUP enthält.

Beispiel: Im folgenden Beispiel wird der Binder angewiesen, das Programm so weit oben wie möglich im Speicher zu plazieren und dann die Adreßoffsets aller Datenelemente in DGROUP so auszurichten, daß sie bei der höchst möglichen Adresse innerhalb der Gruppe geladen werden:

```
link start+prog/high/dsallocate,,,math
```

Anmerkungen:

- Im allgemeinen wird die /DSALLOCATE–Option zusammen mit der /HIGH–Option (s. dort) verwendet, um ungenutzen Speicherplatz vor dem Start des Programms zu nutzen. Der Binder geht davon aus, das alle freien Bytes der Gruppe DGROUP den Speicherplatz vor dem Programm einnehmen. Um diese Gruppe zu nutzen, muß ein Segmentregister auf die Startadresse von DGROUP gesetzt werden.

- Diese Option kann maximal auf /D abgekürzt werden.

/EXEPACK

Format: **/E[XEPACK]**

Bedeutung: Weist den Binder an, Folgen von sich wiederholenden Bytes (i.A. Null–Bytes) zu entfernen und damit die Lade–Verschiebungstabelle zu optimieren, ehe die ausführbare Programmdatei erzeugt wird. Ausführbare Programmdateien, die mit der /EXEPACK–Option erzeugt werden, können kleiner sein und damit schneller geladen werden, als solche, die ohne diese Option gebunden wurden. Das Microsoft–Debug–Programm SYMDEB (s. Handbuch *MASM Reference Guide*) kann jedoch nicht auf Programme, die mit der /EXEPACK–Option gebunden wurden, angewendet werden.

Beispiel: Beim folgenden Beispiel wird eine gepackte Version des Programms PROG.EXE erzeugt:

```
link prog /e;
```

Anmerkungen:

- Mit der /EXEPACK–Option kann nicht in jedem Fall ein namhafter Speicherbereich eingespart werden. In Einzelfällen kann es sogar zu einer Dateivergrößerung kommen. Programme mit einer großen Anzahl von Lade–Verschiebungsvorgängen (etwa 500 oder mehr) und große Folgen sich wiederholender Zeichen werden gewöhnlich durch das Packen mit /EXEPACK kleiner. Wenn Sie nicht sicher sind, ob Ihr Programm diese Bedingungen erfüllt, binden Sie es mit und ohne diese Option und vergleichen Sie das Ergebnis.

- Diese Option kann maximal auf /E abgekürzt werden.

/HELP

Format: **/HE[LP]**

Bedeutung: LINK zeigt eine Liste aller verfügbaren Optionen auf dem Bildschirm an.

Beispiel: Die Eingabe von

```
link /he
```

führt zu folgender Bildschirmanzeige:

```
Microsoft (R) 8086 Object Linker
Version 3.05
Copyright (C) Microsoft Corp. 1983,
1984, 1985. All rights reserved

Valid switches are:

CPARMAXALLOC
DOSSEG
DSALLOCATE
EXEPACK
HELP
HIGH
LINENUMBERS
MAP
NODEFAULTLIBRARYSEARCH
NOGROUPASSICIATION
NOIGNORECASE
OVERLAYINTERRUPT
PAUSE
SEGMENTS
STACK
```

Anmerkungen:

- Diese Option kann maximal auf /HE abgekürzt werden.
- Wenn die /HELP–Option verwendet wird, darf kein Dateiname angegeben werden.

/HIGH

Format: **/H[IGH]**

Bedeutung: Setzt die Startdaresse des gebundenen Programms auf die höchst mögliche Adresse im freien Speicher.

Beispiel: Für das gebundene Programm START.EXE wird die Startadresse auf die höchst mögliche Adresse im freien Speicher gesetzt:

```
link start+prog/h,,;
```

Anmerkungen:

- Normalerweise setzt der Binder die Startadresse des gebundenen Programms auf die niedrigst mögliche Speicheradresse.
- Diese Option kann maximal auf /H abgekürzt werden.

/LINENUMBERS

Format: **/LI[NENUMBERS]**

Bedeutung: Weist den Binder an, die Startadresse jeder Quellprogrammzeile in die List–Datei auszugeben. Die Startadresse einer Quellprogrammzeile ist die Adresse des ersten Befehls in der Objektdatei, der der Quellzeile entspricht. Mit Hilfe des MAPSYSM–Programms aus dem Microsoft–Programmentwicklungssystem MASM (s. Handbuch *MASM Reference Guide*) können die Zeilennummerndaten in eine Symbol–Datei kopiert werden, die dann von SYMDEB benutzt werden kann.

Beispiel: Bei diesem Beispiel werden die Zeilennummern–Informationen der Datei PROG.OBJ in die List–Datei PROG.MAP kopiert:

```
link prog/linenumbers,,math
```

Anmerkungen:

- Der Binder kopiert die Zeilennummern–Informationen nur, wenn in der LINK–Befehlszeile eine List–Datei angegeben wurde und nur, wenn die angegebene(n) Objektdatei(en) Zeilennummerninformationen enthält/enthalten. Die Zeilennumerierung ist in einigen Hochsprachen–Compilern (FORTRAN, PASCAL, C) möglich.

- Der Microsoft–Macroassembler MASM kopiert keine Zeilennummern–Informationen in die Objektdatei.

- Wenn Sie keine List–Datei beim Aufruf des Binders angeben, können Sie dennoch LINK mit der /LINENUMBERS–Option veranlassen, eine zu erzeugen, wenn Sie sie vor oder bei der List–Datei–Anforderung angeben. Der so erzwungenen List–Datei gibt LINK dann den Namen der ersten angegeben Objektdatei und die Namenserweiterung .MAP.

- Diese Option kann maximal auf /LI abgekürzt werden.

/MAP

Format: **/MAP**

Bedeutung: Veranlaßt den Binder, in die List–Datei eine Liste aller als PUBLIC deklarierten Symbole (PUBLIC–Symbole sind für das gesamte Programm allgemein gültige Symbole) auszugeben (s.a. Kapitel 10.2.5). Wenn Sie mit dem SYMDEB–Programm aus dem Microsoft–Makroassembler–Entwicklungssystem arbeiten wollen, müssen Sie beim Binden die /MAP–Option benutzen.

Beispiel: Bei diesem Beispiel wird eine Liste aller PUBLIC–Symbole aus der Objektdatei PROG.OBJ in die List–Datei, die zu diesem Zwecke erzwungen wird, ausgegeben:

```
link prog,,/map;
```

Anmerkungen:

- Wenn Sie keine List–Datei beim Aufruf des Binders angeben, können Sie dennoch LINK mit der /MAP-Option veranlassen, eine zu erzeugen, wenn Sie sie vor oder bei der List–Datei–Anforderung angeben. Der so erzwungenen List–Datei gibt LINK dann den Namen der ersten angegebenen Objektdatei und die Namenserweiterung .MAP.

- Diese Option kann maximal auf /LI abgekürzt werden.

/NODEFAULTLIBRARYSEARCH

Format: **/NOD[EFAULTLIBRARYSEARCH]**

Bedeutung: Weist den Binder an, jegliche Bibliotheksnamen, die er in der/den angegebenen Objektdatei(en) findet, zu ignorieren. Hochsprachen–Compiler können nämlich einen Bibliotheksnamen in eine Objektdatei einfügen, um sicherzustellen, daß ein voreingestellter Satz von Bibliotheken in das Programm eingebunden wird. Mit dieser Option werden solche Namen ignoriert und es wird dem Benutzer überlassen, die einzubindenen Bibliotheken explizit beim LINK–Aufruf anzugeben.

Beispiel: Beim folgenden Beispiel werden die beiden Objektdateien START.OBJ und PROG.OBJ mit Routinen aus den Bibliotheken EM.LIB und MATH.LIB zusammengebunden, wobei jegliche ggf. in den Objektdateien angegebenen Bibliotheken ignoriert werden:

```
link start+prog/nod,,,em+math
```

Anmerkungen:

- Diese Option kann maximal auf /NOD abgekürzt werden.

/NOGROUPASSOCIATION

Format: **/NOG[ROUPASSOCIATION]**

Bedeutung: Weist den Binder an, beim Zuweisen von Adressen zu Code– und Datenelementen Gruppenzusammengehörigkeiten zu ignorieren.

Anmerkungen:

- Diese Option dient nur der Erhaltung der Kompatibilität von Objektdateien, die mit Microsoft–FORTRAN– oder –PASCAL–Versionen 3.13 oder älter erzeugt wurden. Sie sollten diese Option deshalb nur benutzen, wenn Sie solche Objektdateien oder Laufzeitbibliotheken, die zu den genannten älteren Versionen gehören, binden.

- Diese Option kann maximal auf /NOG abgekürzt werden.

/NOIGNORECASE

Format: **/NOI[GNORECASE]**

Bedeutung: Weist den Binder an, bei Symbolnamen zwischen Groß– und Kleinbuchstaben zu unterscheiden.

Beispiel: Im folgenden Beispiel wird die Objektdatei PROG.OBJ mit Routinen aus der Standard–C–Bibliothek SLIB.LIB, die im \LIB–Unterverzeichnis der Disk im aktuellen Laufwerk vorhanden ist, zusammengebunden. Bei der Programmiersprache C wird zwischen Groß– und Kleinschreibung unterschieden:

```
link prog/noi,,,\lib\slib
```

Anmerkungen:

- Normalerweise unterscheidet der Binder bei Symbolnamen **nicht** zwischen Groß– und Kleinbuchstaben. Für ihn sind deshalb die Namen ZWEI, Zwei und zwei dasselbe. Sobald die /NOI–Option verwendet wird, werden diese Namen als drei unterschiedliche Symbole behandelt.
- Üblicherweise wird diese Option beim Binden von mit Hochsprachen–Compilern erzeugten Objektdateien verwendet. Einige Compiler (z.B. der Microsoft–C–Compiler) unterscheiden zwischen Groß– und Klein–Buchstaben bei Symbolnamen und erwarten vom Binder dasselbe.
- Wenn Sie z.B. Module, die mit MASM erzeugt wurden, mit Moduln des C–Compilers zusammenbinden, müssen Sie sicherstellen, daß PUBLIC–Symbole in beiden Modul–Typen dieselbe Unterscheidung zwischen Groß– und Kleinschreibung haben. Sie können z.B. alle Variablen–Namen in einem C–Modul beliebig buchstabieren und dann ohne die /NOI–Option binden. Alternativ können Sie aber auch mit der /MX– oder /ML–Option bei MASM zwischen Groß– und Kleinschreibung bei PUBLIC–Symbolnamen unterscheiden und dann mit der /NOI–Option des Binders binden.
- Diese Option kann maximal auf /NOI abgekürzt werden.

/OVERLAYINTERRUPT

Format: **/O[VERLAYINTERRUPT]:*Zahl***

Bedeutung: Setzt die Interrupt–Nummer des Overlay–Laders auf ***Zahl*** und übergeht damit die Standard–Interrupt–Nummer 03FH.

Zahl Eine ganze Zahl zwischen 0 und 255 in dezimaler, oktaler oder hexadezimaler Codierung. Oktale Zahlen müssen mit einer 0 und hexadezimale mit einer 0 gefolgt vom Kleinbuchstaben **x** (z.B. 0x2B) angegeben werden.

Beispiele: Das erste Beispiel setzt die Interrupt–Nummer des Overlay–Laders auf 255:

```
link prog/o:255,,,math
```

Im zweiten Beispiel wird die Interrupt–Nummer 255 hexadezimal angegeben:

```
link prog/o:0xFF,,,math
```

Im letzten Beispiel wird die Interrupt-Nummer 255 oktal angegeben:

```
link prog/o:0377,,,math
```

Anmerkungen:

- Der Microsoft–Macro–Assembler verfügt nicht über eine Overlay–Verwaltung. Sie können deshalb diese Option nur benutzen, wenn Sie in Ihr Programm einen Laufzeit–Modul von einem Hochsprachen-Compiler einbinden, der Overlays unterstützt. Die jeweiligen Compiler–Handbücher geben einschlägige Auskunft.

- **Achtung:** Verwenden Sie niemals Interruptnummern, die mit den MS–DOS–Standard–Interrupts kollidieren.

- Diese Option kann maximal auf /O abgekürzt werden.

/PAUSE

Format: **/P[AUSE]**

Bedeutung: Weist den Binder an, vor dem Schreiben der ausführbaren Programmdatei zu unterbrechen, damit Disketten gewechselt werden können. Der Binder gibt dann folgende Meldung aus:

```
About to generate .EXE file
Change diskette in drive x and press
<ENTER>
```

(Bereit zur .EXE–Datei–Erzeugung
Diskette in Laufwerk *x* wechseln und Eingabetaste drükken). Für ***x*** wird das Laufwerk angegeben, auf dessen Disk die ausführbare Programmdatei erzeugt wird. Diese Meldung erscheint, nachdem der Binder die angegebenen Objektdateien und Bibliotheken eingelesen hat und die List–Datei, falls angegeben, geschrieben hat. LINK setzt den Bindevorgang fort, sobald die Eingabetaste gedrückt wird und zeigt nach dem Schreiben der ausführbaren Datei die Meldung:

```
Please replace original diskette
in drive x and press <ENTER>
```

(Bitte wieder Originaldiskette in Laufwerk *x* einlegen und die Eingabetaste drücken).

Beispiel: Beim folgenden Beispiel wartet der Binder vor dem Schreiben der ausführbaren Datei PROG.EXE auf Diskettenwechsel. Nach dem Schreiben wartet er wieder, damit die Orignaldiskette wieder eingelegt werden kann:

```
link prog/pause,prog,,\lib\math
```

Anmerkungen:

- **Achtung:** Nehmen Sie nicht die Diskette aus dem Laufwerk, die für die temporäre Arbeitsdatei VM.TMP (s.a. Kapitel 10.2.6) benötigt wird. Sollte bei einem Bindevorgang, der mit der /PAUSE–Option

gestartet wurde, die Meldung erscheinen, daß einen temporäre Arbeitsdatei eingerichtet wurde, müssen Sie den Binder mit Ctrl–Break oder Ctrl–C abbrechen und die Dateien so vorbereiten, daß die temporäre Arbeitsdatei und die ausführbare Programmdatei auf dieselbe Disk geschrieben werden können und starten Sie LINK erneut, diesmal ohne /PAUSE–Option.

- Diese Option kann maximal auf /P abgekürzt werden.

/SEGMENTS

Format: **/SE[GMENTS]:*Zahl***

Bedeutung: Weist den Binder an, nicht mehr als ***Zahl*** Segment pro Programm zu verarbeiten. Trifft LINK mehr als die angegebene Zahl an Segmenten an, wird eine Fehlermeldung angezeigt und der Bindevorgang wird abgebrochen. Diese Option wird benutzt, um den Standardwert von 128 Segmenten zu übergehen.

Zahl Eine ganze Zahl zwischen 1 und 1024 in dezimaler, oktaler oder hexadezimaler Codierung. Oktale Zahlen müssen mit einer 0 und hexadezimale mit einer 0 gefolgt vom Kleinbuchstaben **x** (z.B. 0x2B) angegeben werden.

Beispiele: Es werden 192 Segmente berücksichtigt:

```
link prog/se:192,,;
```

Hier werden 255 Segmente berücksichtigt:

```
link moda+modb,run/segments:0xff,ab;
```

Schließlich noch ein Beispiel für oktale Schreibweise (20 Segmente):

```
link start+prog,/se:024,;
```

Anmerkungen:

- Wird die /SEGMENTS–Option nicht benutzt, weist LINK ausreichend Speicher zu, um bis zu 128 Segmente verarbeiten zu können. Hat Ihr Programm mehr Segmente, müssen Sie die Anzahl maximal zu berücksichtigender Segmente heraufsetzen. Erhalten Sie jedoch die Meldung

  ```
  Segment limit set to high
  ```

  (Segment–Limit zu groß), so müssen Sie die Anzahl maximal zu berücksichtigender Segmente heruntersetzen.

- Diese Option kann maximal auf /SE abgekürzt werden.

/STACK

Format: **/ST[ACK]:*Zahl***

Bedeutung: Setzt den Programmstapelspeicher auf ***Zahl*** Bytes. Der Binder berechnet die Programmstapelspeichergröße normalerweise automatisch aus der Größe aller Stapelsegmente aus den Objektdateien. Dieser berechnete Wert wird durch den bei der /STACK–Option angegebenen ersetzt, wenn diese benutzt wird.

Zahl Eine ganze Zahl zwischen 1 und 65535 in dezimaler, oktaler oder hexadezimaler Codierung. Oktale Zahlen müssen mit einer 0 und hexadezimale mit einer 0 gefolgt vom Kleinbuchstaben **x** (z.B. 0x2B) angegeben werden.

Beispiele: Es werden 512 Bytes für den Stapelspeicher reserviert:

```
link prog/st:512,,;
```

Hier werden 255 Bytes Stapelspeicher reserviert:

```
link moda+modb,run/stack:0xff,ab;
```

Schließlich noch ein Beispiel für oktale Schreibweise (20 Bytes Stapelspeicher):

```
link start+prog,/st:024,;
```

Anmerkungen:

- Mit Hilfe des EXEMOD–Programms kann die Größe des Programmstapelspeichers auch noch nach dem Binden verändert werden.
- Diese Option kann maximal auf /ST abgekürzt werden.

10.4 Wie LINK arbeitet

LINK erzeugt eine ausführbare Programmdatei durch Verknüpfung eines Programmcodesegmentes mit Daten–Segmenten entsprechend den in den Original–Quelldateien (Objekt–Dateien) gegebenen Anweisungen. Diese verknüpften Segmente bilden ein ausführbares Abbild, das direkt in den Speicher kopiert wird, sobald das Programm aufgerufen wird. Die Reihenfolge und Art und Weise, in der LINK die Segmente in die ausführbare Datei kopiert, bestimmt die Reihenfolge und Art und Weise, in der der Lader die Segmente in den Speicher lädt.

Sie können den Binder anweisen, wie er die Segmente binden soll, indem Sie mit Hilfe der SEGMENT–Direktive des Assemblers oder Hochsprachencompilers Segment–Attribute definieren oder indem Sie mit Hilfe der GROUP–Direktive Segment–Gruppen bilden. Diese Direktiven definieren Gruppen–Zusammenhänge und Klassen, und kombinieren Elemente, die die Ordnung und relativen Startadressen aller Segmente in einem Programm festlegen. Diese Maßnahmen sind von jenen, die durch Angabe von Optionen in der LINK–Befehlszeile ausgelöst werden, unabhängig.

In den folgenden Abschnitten wird beschrieben, wie LINK die Segmente zusammenfügt und Referenzen auf Elemente im Speicher auflöst.

10.4.1 Segment–Ausrichtung

Der Binder richtet die Startadresse eines jeden Segmentes nach Bytes, Worten (Vielfache von 2 Bytes), Paragraphen (Vielfache von 16 Bytes) oder Seiten (Vielfache von 256 Bytes) aus. Voreinstellungsmäßig wird nach Paragraphen ausgerichtet.

Wenn der Binder ein Segment findet, prüft er zunächst, nach welcher Grenze er dieses ausrichten soll, ehe er das Segment in die ausführbare Programmdatei kopiert. Soll nach Wort, Paragraph oder Seite ausgerichtet werden, so untersucht der Binder den bereits gebundenen Programmteil, ob das letzte kopierte Byte an einer entsprechenden Grenze liegt. Falls nicht, wird der fehlende Bereich mit Null–Bytes aufgefüllt.

10.4.2 Rahmenzahl

Der Binder berechnet für jedes Programmsegment eine Startadresse. Diese Adresse basiert auf der Segmentausrichtungs–Grenze (s. o.) und der Größe der bereits gebundenen Segmente. Die Adresse besteht aus einem Offset und einer kanonischen Rahmenzahl, die ihrerseits die Adresse des ersten Paragraphen im Speicher repräsentiert, der ein oder mehrere Bytes des Segmentes enthält. Die Rahmenzahl ist immer ein Vielfaches von 16, also einer Paragraphenadresse und der Offset repräsentiert die Anzahl Bytes vom Anfang des Paragraphen zum ersten Byte des Segmentes. Bei der Segmentausrichtung nach Bytes oder Worten kann der Offset von Null verschieden sein, bei Paragraphen– und Seitenausrichtung ist er dagegen immer Null.

Die Rahmenzahl eines Segmentes kann aus der gebundenen Datei ermittelt werden. Sie besteht aus den ersten fünf hexadezimalen Ziffern der für das Segment angegebenen Startadresse.

10.4.3 Segment–Reihenfolge

Der Binder kopiert die Segmente in derselben Reihenfolge in die ausführbare Programmdatei, wie er sie in den Objekt–Dateien auffindet. Diese Ordnung wird solange beibehalten, bis der Binder zwei oder mehr Segmente mit demselben Klassen–Namen findet. Segmente mit identischem Klassen–Namen gehören zum selben Klassen–Typ und werden als zusammenhängende Blöcke in die ausführbare Programmdatei kopiert.

Das Handbuch *Microsoft Macroassembler Reference Manual* beinhaltet eine detailliertere Diskussion über die Segment–Ladeordnung und Möglichkeiten der Beeinflussung der Ladeordnung durch die Zuweisung von Klassentypen.

10.4.4 Kombinierte Segmente

Der Binder bestimmt mit Hilfe von Kombinationstypen, ob zwei oder mehrere Segmente, die sich denselben Namen teilen, zu einem großen Segment zusammengefaßt werden sollen. Diese Kombinationstypen sind PUBLIC, STACK, COMMON, MEMORY, AT und PRIVATE. Auch diese Typen sind in dem Handbuch *Microsoft Macroassembler Reference Manual* beschrieben.

Hat ein Segment den Kombinationstyp PUBLIC, so ordnet es der Binder automatisch den anderen Segmenten zu, die denselben Namen haben und zur selben Klasse gehören. Bei der Kombination von Segmenten stellt der Binder sicher, daß die Segmente aufeinander folgen und daß auf alle Adressen solcher Segmente mit einem Offset von derselben Rahmenadresse aus zugegriffen werden kann. Das Ergebnis ist dann dasselbe, als wenn das Segment als ein Block in der Quelldatei definiert worden wäre.

Der Binder merkt sich den Typ der Ausrichtung für jedes Segment. Das bedeutet, daß, wenn auch die Segmente zu einem einzelnen großen Segment gehören, Code und Daten in den einzelnen Segmenten ihren ursprünglichen Ausrichtungstyp beibehalten. Wenn eine Segmentkombination größer als 64 Kbytes wird, meldet der Binder einen Fehler.

Hat ein Segment den Kombinationstyp STACK, verfährt der Binder genauso wie bei den PUBLIC–Segmenten. Der einzige Unterschied bei den STACK-Segmenten ist der, daß der Binder hier einen Stapelzeigerwert in die ausführbare Programmdatei einträgt. Dieser Wert ist ein Offset zum Ende des ersten Stapelsegmentes oder kombinierten Stapelsegmentes, das der Binder antrifft.

Wird der STACK–Kombinationstyp für Stapelsegmente verwendet, sind keine gesonderten Befehle zum Laden der Segmentadresse in das SS–Register des Prozessors erforderlich.

Hat ein Segment den Kombinationstyp COMMON, so ordnet es der Binder automatisch den anderen Segmenten zu, die denselben Namen haben und zur selben Klasse gehören. Bei der Kombination von COMMON–Segmenten plaziert er jedoch den Anfang jedes Segmentes bei derselben Adresse und erzeugt so eine Reihe von sich überlappenden Segmenten. Das Ergebnis ist dann ein einzelnes Segment, das nicht größer ist als das größte der so kombinierten Einzelsegmente.

Hat ein Segment den Kombinationstyp MEMORY, verfährt der Binder genauso wie bei den PUBLIC–Segmenten. Der Microsoft–Macro–Assembler unterstützt diesen Typ nur, um die Kompatibilität zu Bindern zu wahren, die einen separaten Kombinationstyp für MEMORY–Segmente unterstützen.

Ein Segment hat den Kombinationstyp PRIVATE nur dann, wenn dafür kein expliziter Kombinationstyp in der Objektdatei definiert ist. PRIVATE–Segmente werden von LINK nicht kombiniert.

10.4.5 Gruppen

In Gruppen können nicht aufeinanderfolgende Segmente, die nicht zur selben Klasse gehören, relativ zur selben Rahmenadresse adressiert werden. Wenn der Binder eine Gruppe antrifft, werden alle Speicherreferenzen auf Elemente in der Gruppe so ausgerichtet, daß sie relativ zur selben Rahmenadresse liegen.

Segmente innerhalb einer Gruppe brauchen nicht aufeinander zu folgen, nicht zur selben Klasse zu gehören und auch nicht denselben Kombinationstyp haben. Die einzige Bedingung ist, daß alle Segmente einer Gruppe in einen Speicherbereich von 64 KBytes passen müssen.

Gruppen beeinflussen die Segment–Ladeordnung nicht. Solange Sie nicht Klassen–Namen verwenden und die Objektdateien in der richtigen Reihenfolge angeben, garantiert der Binder nicht die Aufeinanderfolge dieser Segmente. Er kann durchaus Segmente, die nicht zu der Gruppe gehören, in denselben 64–KBytes–Speicherbereich plazieren. Obwohl der Binder nicht explizit prüft, ob alle Segmente einer Gruppe in einen 64–KBytes–Speicherbereich passen, ist es sehr wahrscheinlich, daß er die Meldung

```
Fixup overflow error
```

(Überlauf bei der Segmentfixierung) anzeigt, wenn diese Bedingung nicht erfüllt ist.

Gruppen und ihre Definition werden detailliert im Handbuch *Microsoft Macroassembler Reference Manual* beschrieben.

10.4.6 Fixierung aller nichtzugeordneten Referenzen

Sobald die Startadresse jedes Segmentes in einem Programm bekannt ist und alle Segment–Kombinationen und Gruppen festgelegt sind, kann der Binder alle nichtzugeordneten Referenzen zu Marken und Variablen fixieren. Dazu berechnet der Binder einen geeigneten Offset und eine geeignete Segmentadresse und ersetzt damit die temporären, vom Assembler oder Compiler erzeugten, Werte.

LINK führt solche Fixierungen für vier verschiedenen Referenzen durch:

- kurze Referenzen
- nahe selbst–relative Referenzen
- nahe segment–relative Referenzen
- lange Referenzen

Die Größe des zu berechnenden Wertes für die Fixierung hängt vom Referenz–Typ ab. Findet der Binder einen Fehler in der erwarteten Größe einer Referenz, wird die schon erwähnte Fehlermeldung

```
Fixup overflow error
```

angezeigt. Dies kann z.B. passieren, wenn in einem Programm versucht wird, mit einem 16–Bit–Offset zu einem Befehl zu gelangen, der in einem Segment mit einer anderen Rahmenadresse liegt.

Kurze Referenzen sind in JMP–Befehlen enthalten, bei denen die Programmsteuerung an einen markierten Befehl übergeben wird, der in demselben Segment oder derselben Gruppe liegt. Das Sprungziel darf dabei nicht mehr als 128 Bytes vom Ort der Referenz entfernt sein. Der Binder berechnet einen vorzeichenbehafteten 8–Bit–Wert für diese Referenz und zeigt eine Fehlermeldung an, wenn das Sprungziel zu einem anderen Segment oder einer anderen Gruppe gehört oder weiter als 128 Bytes entfernt ist.

Nahe selbst-relative Referenzen sind in Befehlen enthalten, die auf Daten relativ zum selben Segment oder zur selben Gruppe zugreifen. Der Binder berechnet in diesem Fall einen 16-Bit-Offset für diese Referenz und zeigt einen Fehler an, wenn die bezogenen Daten nicht im selben Segment oder derselben Gruppe liegen.

Nahe segment-relative Referenzen sind in Befehlen enthalten, die auf Daten in einem angegebenen Segment oder einer angegebenen Gruppe zugreifen oder die relativ zu einem angegebenen Segmentregister liegen. LINK berechnet in diesem Fall einen 16-Bit-Offset und zeigt einen Fehler an, wenn eine der folgenden Bedingungen erfüllt ist:

- Der Offset des Zieles innerhalb des angegebenen Rahmens ist größer als 64 KBytes oder kleiner als 0.
- Der Anfang des kanonischen Rahmens des Ziels ist nicht adressierbar.

Lange Referenzen kommen in CALL-Befehlen vor, die sich auf einen Befehl in einem anderen Segment oder einer anderen Gruppe beziehen. LINK berechnet hier eine 16-Bit-Rahmenadresse und einen 16-Bit-Offset und zeigt eine Fehlermeldung an, wenn eine der folgenden Bedingungen erfüllt ist:

- Der berechnete Offset ist größer als 64 KBytes oder kleiner als 0.
- Der Anfang des kanonischen Rahmens des Ziels ist nicht adressierbar.

11. Das Testhilfeprogramm DEBUG

In diesem Kapitel lernen Sie, wie

- das Testhilfeprogramm DEBUG gestartet wird;
- die DEBUG–Befehle und Befehlserweiterungen zum Programmtest eingesetzt werden können.

11.1 Einführung

DEBUG ist ein Testhilfeprogramm, das eine benutzer–geführte Testumgebung für binäre und ausführbare Objektdateien zur Verfügung stellt. Wie EDLIN, der MS–DOS–Zeileneditor, zum Bearbeiten und Ändern von Textdateien dient, so erlaubt DEBUG das Bearbeiten und Ändern von Binär–(Programm–)Dateien.

DEBUG macht die Neuassemblierung eines Programms, das geringfügig geändert wurde, überflüssig. Es erlaubt die Änderung des Inhaltes einer Datei oder eines Mikroprozessor–Registers und die sofortige Neuausführung des so geänderten Programmes, um die Wirksamkeit der Änderungen zu überprüfen.

Die Ausführung jedes DEBUG–Befehls kann zu jeder Zeit durch Drücken von Ctrl–Break oder Ctrl–C abgebrochen werden. Mit Ctrl–S kann die Bildschirmausgabe angehalten werden, um lange Ausgaben bildschirmweise lesen zu können, und dann mit jeder anderen Taste fortgesetzt werden. Diese Steuertastenfunktionen sind kompatibel zu jenen auf der MS–DOS–Befehlsebene.

11.2 DEBUG starten

DEBUG kann auf zweierlei Weise gestartet werden. Bei der ersten Methode wird nur der Programmname, also DEBUG, eingegeben. Bei der zweiten Methode wird zusätzlich ein Dateiname sowie ggf. eine Argumentliste angegeben, also:

Methode 1: **DEBUG**

Methode 2: **DEBUG** [*Datei*[*.Erw*][*Argliste*]]

Beide Methode werden im folgenden beschrieben:

Methode 1

Um DEBUG zu starten, geben Sie nur ein:

```
debug
```

Das Testhilfeprogramm antwortet mit seiner Bereitschaftsanzeige, einem Bindestrich (–), und wartet auf die Eingabe von Befehlen. Da kein Dateiname angegeben wurde, kann auf der DEBUG–Befehlsebene mit dem aktuellen Speicher, Disk–Sektoren oder –Dateien gearbeitet werden.

Achtung: Wenn DEBUG (Version 2.0) gestartet wird, richtet es sich einen Programmkopf beim Offset 0 im Programm–Arbeitsbereich ein. In früheren DEBUG–Versionen konnte dieser Kopf überschrieben werden. Der Standardkopf kann auch bei dieser Version noch überschrieben werden, wenn DEBUG ohne Dateiangabe aufgerufen wurde. Wenn dagegen beim Aufruf eine .COM– oder .EXE–Datei angegeben wurde, darf der Programmkopf unterhalb der Adresse 5CH nicht verändert werden. Andernfalls wird DEBUG beendet.

Vermeiden Sie den Neustart eines Programms, nachdem die Meldung

```
Programm terminated normally
```

(Programm normal beendet) angezeigt wurde. Statt dessen müssen Sie das Programm mit den Befehlen **N** (Name) und **L** (Lade) (s. dort in Kapitel 11.5) neu laden, um es erneut starten zu können.

Methode 2:

Alternativ können Sie DEBUG auch mit einer Befehlszeile aufrufen. Die allgemeine Syntax lautet dann:

[*L*:][*BPfad*]**DEBUG** [[*Laufw*:][*Pfad*][*Datei*[*.Erw*]][*Argliste*]]

Dabei bedeuten:

L*:*BPfad Laufwerk und Suchpfad für die DEBUG–Programmdatei.

Laufw:PfadDatei.Erw Die vollständige Angabe für die nach dem Aufruf von DEBUG zu ladende Binärdatei. Wird ***Laufw:*** weggelassen, so nimmt DEBUG das aktuelle Laufwerk an. Wird ***Pfad*** weggelassen, so nimmt DEBUG das aktuelle Verzeichnis auf der Disk im angegebenen Laufwerk an.

Argliste Die Liste von Argumenten oder Parametern, mit denen das zu testende Programm normalerweise aufgerufen würde.

Beispiel: Im Falle von DISKCOPY würde der Aufruf mit DEBUG lauten:

```
debug diskcopy a: b:
```

11.3 Befehlsstruktur

Jeder DEBUG–Befehl besteht aus einem einzelnen Buchstaben, gefolgt von einem oder mehreren Parametern.

Alle Steuerzeichen und die speziellen MS–DOS–Edier– und Funktionstasten, wie sie in Kapitel 7 beschrieben wurden, gelten auch bei der Benutzung von DEBUG.

Wenn ein Fehler bei einem DEBUG–Befehl auftritt, wird die Befehlszeile noch einmal angezeigt und in der folgenden Zeile erscheint als Fehlerindikator das Zeichen ^ gefolgt vom Wort 'Error' (Fehler), also z.B.:

```
dcs:100 cs:110
     ^ Error
```

Bei der Eingabe von Befehlen und Parametern ist jede Kombination von Groß– und Kleinbuchstaben zulässig. DEBUG wandelt intern alle Buchstaben grundsätzlich in Großbuchstaben um.

Die folgende Tabelle enthält alle DEBUG–Befehle in einer Übersicht:

DEBUG–Befehl	Funktion
A [*Adresse*]	Assembliere
C *Bereich Adresse*	Vergleiche
D [*Bereich*]	Zeige Speicherinhalt
E *Adresse* [*Liste*]	Ändere Speicherinhalt
F *Bereich Liste*	Fülle Speicher
G [=*Adresse*[*Adresse*. . .]]	Gehe zu
H *Wert Wert*	Hex–Arithmetik
I *Wert*	Eingabe vom Port
L [*Adresse*[*Laufw:Satz Satz*]]	Lade
M *Bereich Adresse*	Verschiebe
N *Datei*[*Datei*]	Setze Namen
O *Wert Byte*	Ausgabe über Port
Q	Beenden
R [*Register*]	Zeige Registerinhalt
S *Bereich Liste*	Suche
T [=*Adresse*] [*Wert*]	Verfolge Programmlauf
U [*Bereich*	Disassembliere
W [*Adresse*[*Laufw:Satz Satz*]]	Schreibe auf Disk

11.4 Befehlsparameter

Alle DEBUG–Befehle mit Ausnahme des Beendigungsbefehls **Q** akzeptieren Parameter. Mehrere Parameter können durch Trennzeichen (Leerstelle oder Komma (,)) voneinander getrennt werden. Benötigt wird ein Trennzeichen allerdings nur zwischen zwei aufeinanderfolgenden hexadezimalen Werten. Folgende Befehle sind deshalb gleich und alle gültig:

```
dcs:100 110
d cs:100 110
d,cs:100,110
```

Im folgenden werden die in der Befehlsübersicht im letzten Abschnitt angegebenen Parameter beschrieben.

Adresse Ein zweiteiliger Parameter, der entweder aus einer alphabetischen Segmentregisterangabe oder einer vierstelligen Segmentadresse sowie einem Offset–Wert besteht. Segmentregister oder Segmentadresse können weggelassen werden. In diesem Fall wird für alle Befehle außer **G, L, T, U** und **W** voreinstellungsmäßig das Datensegmentregister **DS** verwendet. Für die genannten Ausnahmen wird voreinstellungsmäßig das Codesegmentregister **CS** benutzt. Alle numerischen Adreßwerte müssen hexadezimal angegeben werden. Gültige Adreßangaben sind z.B.

```
CS:0100
04BA:0100
```

Beachten Sie, daß zwischen der Segmentregister– oder –adreßangabe ein Doppelpunkt (:) erforderlich ist.

Bereich Ein Parameter, der aus zwei Adressen besteht und die allgemeine Form hat:

Adresse Adresse oder
Adresse **L** *Wert*

Wert gibt die Anzahl von Zeilen an, auf die der Befehl wirken soll, und muß als ganze Hexadezimalzahl angegeben werden. Die zweite Syntax kann nicht benutzt werden, wenn ein weiterer Hexadezimalwert folgt, denn dieser würde als zweite Adreßangabe im *Bereich* interpretiert. Hier drei Beispiele für gültige Bereichsangaben:

```
CS:100 110
CS:100 L 10
CS:100
```

Nicht erlaubt ist dagegen folgende Bereichsangabe:

```
CS:100 CS:100
^ Error
```

Bereich ist auf 10000H begrenzt. Soll dieser Wert benutzt werden, so muß 0000 oder 0 angegeben werden.

Byte Ein zweistelliger Hexadezimalwert, der zu schreibende oder zu lesende Daten spezifiziert.

Datei Eine gültige Dateibezeichnung nach MS–DOS–Konventionen in der Form

[*Laufw*:][*Pfad*]*Dateiname*[.*Erw*]

Wird *Laufw*: weggelassen, so nimmt DEBUG das aktuelle Laufwerk an. Wird *Pfad* weggelassen, so nimmt DEBUG das aktuelle Verzeichnis der Disk im angegebenen Laufwerk an. Für *.Erw* existiert keine Voreinstellung.

Laufw: Ein einstelliger Hexadezimalwert, der angibt, von welchem Laufwerk eine Datei gelesen oder auf welches Laufwerk eine Datei geschrieben werden soll. Erlaubt sind die Werte 0 bis 3 mit der Zuordnung 0 = A, 1 = B, 2 = C, 3 = D.

Liste Eine Folge von *Byte*– oder *String*–Werten. *Liste* muß der letzte Parameter in einem Befehl sein. Hier ein Beispiel für eine Liste beim **F**–Befehl:

```
fcs:100 42 45 52 54 2E
```

Register Siehe R–Befehl.

Satz Ein ein– bis dreistelliger Hexadezimalwert, der die logische Satznummer auf der Disk bzw. die Anzahl der zu lesenden oder zu schreibenden Disk–Sektoren angibt. Logische Sätze entsprechen dabei physikalischen Disksektoren.

String Eine beliebige Anzahl von Zeichen, eingeschlossen in Apostrophs (') oder Anführungszeichen ("). Wenn die Begrenzungszeichen selbst Bestandteil des Strings sind, müssen Anführungszeichen verwendet werden. Hier Beispiele für gültige und ungültige String–Angaben:

```
'Diese "Zeichenkette" ist gültig'
'Diese ''Zeichenkette'' ist gültig'
'Diese 'Zeichenkette' ist ungültig'
'Diese ""Zeichenkette"" ist ungültig'
"Diese 'Zeichenkette' ist gültig"
"Diese ""Zeichenkette"" ist gültig"
"Diese "Zeichenkette" ist ungültig"
"Diese ''Zeichenkette'' ist ungültig"
```

DEBUG benutzt bei einem String die ASCII–Codes der einzelnen Zeichen als Liste von Byte–Werten.

Wert Ein hexadezimaler Wert mit bis zu vier Stellen, der entweder die Adresse einer Ein–/Ausgabe–Schnittstelle (Port–Adresse) oder die Anzahl von zu wiederholenden Befehlsausführungen angibt.

11.5 Die DEBUG–Befehle im einzelnen

Im folgenden wird jeder DEBUG–Befehl detailliert beschrieben. Das Format und die Syntax der Beschreibungen entsprechen denen bei der Beschreibung der MS–DOS–Befehle (s.a. Kapitel 6.2). Jede einzelne Befehlsbeschreibung beginnt auf einer neuen Seite.

A (Assembliere)

Format: **A** [***Adresse***]

Bedeutung: Assembliert mnemonischen 8086/8087/8088–Code direkt in den Speicher. Wird ***Adresse*** nicht angegeben, so wird ab Adresse CS:0100 assembliert, wenn vorher noch kein **A**–Befehl benutzt wurde, andernfalls bei der beim letzten benutzen **A**–Befehl angegebenen Adresse. Eine Leereingabe beendet den **A**–Befehl.

Anmerkungen:

- Wird während der Assemblierung ein Syntaxfehler gefunden, antwortet DEBUG mit der Meldung

  ```
  ^ Error
  ```

  und zeigt die aktuelle Assemblierungsadresse wieder an.

- Alle numerischen Werte sind grundsätzlich in hexadezimaler Schreibweise mit 1 bis 4 Stellen anzugeben.

- Mnemonische Präfixe müssen **vor** dem Befehl angegeben werden, auf den sie sich beziehen, können jedoch in einer eigenen Zeile stehen.

- Segmente werden durch die mnemonischen Codes CS:, DS:, ES: und SS: bezeichnet.

- Der mnemonische Code für den fernen Rücksprung (far return) lautet RETF.

- Bei den mnemonischen Codes für Zeichenkettenmanipulationen muß die Zeichenkettengröße explizit angegeben werden, also z.B. bei der Verschiebung von Wortzeichenketten MOVSW und bei der Verschiebung von Byte–Zeichenketten MOVSB.

- Der Assembler assembliert automatisch kurze, nahe oder ferne Sprünge und Unterprogrammaufrufe ab-

hängig von der relativen Byteadresse in die Zieladresse. Solche Sprünge können durch Verwendung des Präfixes NEAR (nah) oder FAR (fern) überschrieben werden, wie das folgende Beispiel zeigt:

```
0100:0500  JMP 502          ;2-Byte-Sprung
                            (kurz)
0100:0502  JMP NEAR 505     ;3-Byte-Sprung
                            (nah)
0100:505   JMP FAR 50A      ;5-Byte-Sprung
                            (fern)
```

Das NEAR–Präfix kann mit NE abgekürzt werden. Für FAR gibt es keine Abkürzung.

- DEBUG kann nicht unterscheiden, ob ein Operand sich auf eine Wort– oder Byte–Speicheradresse bezieht. In einem solchen Fall muß der Datentyp explizit mit dem Präfix WORD PTR oder BYTE PTR angegeben werden (Abkürzungen: WO und BY). Z.B.:

```
NEG BYTE PTR [128]
DEC WO[SI]
```

- DEBUG kann auch nicht unterscheiden, ob sich ein Operand auf einen Speicherplatz oder auf einen Direktoperanden bezieht. Dehalb benutzt DEBUG die Assembler–Konvention, daß sich in eckige Klammern eingeschlossene Operanden auf Speicherplätze beziehen, also z.B.:

```
MOV AX,21   ;lade AX mit 21H
MOV AX,[21] ;lade AX mit dem Inhalt
            ;von Speicherplatz 21H
```

- Der **A**–Befehl von DEBUG unterstützt auch die beiden Pseudo–Befehle DB und DW. DB assembliert Byte–Werte und DW Wort–Werte direkt in den Speicher, wie folgende Beispiele zeigen:

```
DB 1,2,3,4,"DIES IST EIN BEISPIEL"
DB 'DIES IST EIN ANFUEHRUNGSZEICHEN:"'
DB "DIES IST EIN APOSTROPH:'"

DW 1000,2000,3000,"BACH"
```

- Außerdem werden sämtliche Formen der indirekten Registerbefehle unterstützt:

```
ADD BX,34[BP+2].[SI-1]
POP [BP+DI]
PUSH [SI]
```

- Alle Operationscode–Synonyme werden ebenfalls unterstützt:

```
LOOPZ 100
LOOPE 100

JA 200
JNBE 200
```

- Für 8087–Befehle müssen die Präfixe WAIT und FWAIT explizit angegeben werden:

```
FWAIT FADD ST,ST(3) ;hier wird ein
                    ;FWAIT-Präfix assembliert
LD TBYTE PTR [BX]  ;hier nicht
```

C (Vergleiche)

Format: **C *Bereich Adresse***

Bedeutung: Vergleicht den mit ***Bereich*** angegebenen Speicherbereich mit einem gleich großen Speicherbereich, der bei ***Adresse*** beginnt. Sind beide Bereiche identisch, so zeigt der **C**–Befehl nichts an. Werden jedoch Unterschiede gefunden, so werden diese in der folgenden allgemeinen Form angezeigt:

Adresse1 Byte1 Byte2 Adresse2

Die ersten beiden Angaben beziehen sich auf das betreffende Byte in ***Bereich***, die letzten beiden auf das Byte in dem Bereich, der bei ***Adresse*** beginnt.

Beispiel: Die beiden folgenden Befehle bewirken dasselbe, nämlich den Vergleich eines Bereiches von 256 Bytes Länge im Adreßbereich 100H bis 1FFH mit einem anderen, der bei der Adresse 300H beginnt:

```
C100,1FF 300
```
oder

```
C100L100 300
```

D (Zeige Speicherinhalt)

Format: **D** [***Bereich***]

Bedeutung: Zeigt den Inhalt des angegebenen Speicherbereiches auf dem Bildschirm an. Wird ***Bereich*** weggelassen, so werden die ersten 128 Bytes ab der Adresse angezeigt, die auf das letzte mit dem letzten **D**–Befehl angezeigte Byte folgt. Wurde vorher noch kein **D**–Befehl benutzt, so werden 128 Bytes ab der Adresse DS:100 angezeigt.

Beispiele: Die Eingabe des Befehls

```
dcs:100 110
```

führt z.B. zu der Anzeige

```
04BA:0100 42 45 52 . . . 4E 44 TOM SAWYER
```

Wenn nur **D** eingegeben wird, wird die Anzeige in der o.a. Weise formatiert. Jede Zeile beginnt mit einer gegenüber der vorhergehenden Zeile um 16 Bytes höheren Adresse. Jeder weitere **D**–Befehl ohne Parameter zeigt die unmittelbar auf die zuletzt angezeigten Bytes folgenden.

Wenn Sie 32 Bytes im o.a. Anzeigeformat sehen wollen, müssen Sie eingeben:

```
DCS:100 L 20
```

Wenn Sie dann den Befehl

```
dcs:100 115
```

eingeben, erhalten alle Bytes im Adreßbereich 100H bis 115H im CS–Segment angezeigt.

Anmerkungen:

- Die Speicherinhaltsanzeige besteht aus zwei Teilen: einer hexadezimal–codierten Anzeige links und einer ASCII–codierten Anzeige rechts. Nicht darstellbare Codes (00H–1FH und 80H–FFH) werden in der ASCII–Azeige durch einen Punkt (.) dargestellt.

- Jede Anzeigezeile enthält die Inhalte von 16 Bytes mit einem Bindestrich (–) zwischen dem 8. und 9. Byte in der hexadezimalen Anzeige.

- Jede Anzeigezeile beginnt grundsätzlich an einer Paragraph–Grenze (16–Byte–Grenze) und enthält ggf. Leerstellen bis zum ersten Byte des anzuzeigenden Bereiches.

E (Ändere Speicherinhalt)

Format: **E** ***Adresse*** [***Liste***]

Bedeutung: Ersetzt alle Bytes ab ***Adresse*** durch die Bytes aus ***Liste***. Wird ***Liste*** nicht angegeben, so wird der Inhalt des durch ***Adresse*** bezeichneten Speicherplatzes angezeigt und der **E**–Befehl wartet auf Eingabe. Jetzt können folgende Eingaben gemacht werden:

Geben Sie einfach den Wert ein, der den alten Inhalt des angezeigten Speicherplatzes ersetzen soll. Der **E**–Befehl prüft dabei jedes eingegebene Zeichen auf gültigen Hexadezimalcode und lehnt ungültige Eingaben ab.

Drücken Sie die Leertaste, um zum nächsten Speicherplatz zu gelangen. Hier können Sie in der o.a. Weise wieder ändern. In jeder Zeile können Sie so maximal 8 Bytes ändern. Drücken Sie nach dem 8. Byte die Leertaste, wird automatisch eine neue Zeile begonnen mit dem entsprechend Adreß–Präfix.

Drücken Sie den Bindestrich (–), um zum vorhergehenden Speicherplatz zu gelangen. Auch in diesem Fall wird automatisch eine neue Zeile begonnen mit dem entsprechend Adreß–Präfix.

Drücken Sie die Eingabetaste, um den **E**–Befehl zu beenden. Die Eingabetaste kann an jeder Position gedrückt werden.

Beispiele: Angenommen, Sie haben den folgenden Befehl eingegeben:

```
ECS:100
```

und DEBUG bringt folgende Anzeige:

```
04BA:0100 EB._
```

Um diesen Wert auf z.B. 41H zu ändern, geben Sie an der Cursor–Position (_) 41 ein:

```
04BA:0100 EB.41
```

Um zu einem weiter hinten liegenden Speicherplatz zu gelangen, drücken Sie jetzt dreimal die Leertaste. Sie erhalten z.B.:

```
04BA:0100 EB.41      10.   00.   BC._
```

Um den Inhalt BCH auf 42H zu ändern, geben Sie ein:

```
04BA:0100 EB.41      10.   00.   BC.42_
```

Wenn Sie jetzt zurückgehen wollen zum Byte mit dem Inhalt 10H (bei Adresse 0101H), um dessen Inhalt auf 6FH zu ändern, müssen Sie zweimal den Bindestrich (–) drücken und dann 6FH eingeben:

```
04BA:0100 EB.41      10.   00.   BC.42-
04BA:0102 00.-
04BA:0101 10.6F_
```

Drücken der Eingabetaste beendet schließlich den **E**–Befehl und bringt Sie zurück auf die DEBUG–Befehlsebene.

F (Fülle Speicher)

Format: **F *Bereich Liste***

Bedeutung: Füllt den angegebenen Speicherbereich mit den Werten aus ***Liste***. ***Liste*** wird ggf. so oft wiederholt, bis der angegebene Bereich gefüllt ist.

Beispiel: Wenn Sie folgenden Befehl eingeben:

```
F04BA:100 L 100 42 45 52 54 41
```

wird der Speicherbereich zwischen 04BA:100 und 04BA:1FF mit den angegebenen Bytes gefüllt. Die 5 Werte werden so oft wiederholt, bis der Bereich gefüllt ist.

Anmerkungen:

- Wenn ***Liste*** mehr Werte enthält, als ***Bereich*** Bytes, so werden die überschüssigen Werte aus ***Liste*** ignoriert. Bezieht sich ***Bereich*** auf fehlerhaften oder nichtexistierenden Speicher, so wird ein Fehler angezeigt.

G (Gehe zu)

Format: **G** [=***Adresse***[***Adresse*** . . .]]

Bedeutung: Startet ein im Speicher befindliches Programm am Anfang oder ab einer angegebenen Adresse. Wird **G** ohne Parameter benutzt, so wird das aktuell im Speicher befindliche Programm so gestartet, als wäre es von der MS–DOS–Befehlsebene aus gestartet worden. Wird =***Adresse*** angegeben, so beginnt die Programmausführung bei dieser Adresse. Werden weitere Adressen angegeben, so werden diese Adressen als Unterbrechungspunkte interpretiert und die Programmausführung wird bei der Adresse in dieser Liste unterbrochen, die als nächste auf die Startadresse folgt, unabhängig von ihrer Position in der Liste.

Adresse Eine Adresse im Format, wie in Kapitel 11.4 beschrieben. Die erste Adreßangabe nach dem Gleichheitszeichen (=) definiert die Startadresse für das auszuführende Programm. Adreßangaben ohne Gleichheitszeichen(=) stellen Unterbrechungspunkte dar.

Beispiel: Angenommen, Sie haben den folgenden Befehl eingegeben:

```
gcs:7550
```

Das gegenwärtig im Speicher befindliche Programm würde bei seiner normalen Startadresse gestartet und bis zur Adresse 7550H im Codesegment ausgeführt, dann unterbrochen und alle Registerinhalte und Prozessorflags würden angezeigt (s.a. R–Befehl).

Anmerkungen:

- Wenn die Programmausführung bei einem im **G**–Befehl gesetzten Unterbrechungspunkt anlangt, werden alle Register und Flags des Prozessors angezeigt, als ob der **R**–Befehl (s. dort in diesem Kapitel) gegeben worden wäre.

- Es können bis zu zehn Unterbrechungspunkte gesetzt werden, jedoch nur bei Adressen, die das erste Byte eines gültigen 8086/8087–Befehlscodes enthalten. Werden mehr als 10 Unterbrechungspunkte gesetzt, meldet DEBUG den BP–Fehler (s. Kapitel 11.6).

- Der Benutzer–Stapelzeiger muß für diesen Befehl gültig sein und 6 Bytes verfügbar halten. Der **G**–Befehl benutzt den IRET–Prozessorbefehl, um in das zu testende Programm zu springen. Der Benutzer–Stapelzeiger wird damit gesetzt und die Benutzerflags, das Codesegmentregister sowie der Befehlszähler werden auf dem Stapel abgelegt. Ist der Benutzerstapel nicht gültig oder zu klein, kann es zum Systemabsturz kommen. Bei der/den angegebenen Unterbrechungsadresse(n) wird ein Unterbrechungscode (0CCH) gesetzt.

- Wenn DEBUG bei einer Adresse mit dem Unterbrechungscode anlangt, trägt es bei allen Unterbrechungspunkten wieder die Originalbefehle ein, wenn hier die Programmausführung beendet wird.

- Wird nach dem Erreichen eines Unterbrechungspunktes der **G**–Befehl ohne Parameter erneut gegeben, so wird das Programm mit dem nächsten auf die Unterbrechungsadresse folgenden Befehl fortgesetzt, allerdings so, als wäre es unter MS–DOS aufgerufen worden.

H (Hex–Arithmetik)

Format: **H** ***Wert Wert***

Bedeutung: Addiert und subtrahiert die beiden angegebenen Hexadezimalwerte und zeigt die beiden Ergebnisse an.

Wert Zwei Hexadezimalwerte, die addiert bzw. subtrahiert werden sollen. Bei der Subtraktion wird der zweite vom ersten Wert subtrahiert.

Beispiel:

```
H19F 10A
02A9  0095
```

I (Eingabe vom Port)

Format: **I** ***Wert***

Bedeutung: Liest ein Byte über die angegebene Ein–/Ausgabeschnittstelle und zeigt den Wert an.

Wert Eine gültige 16–Bit–Adresse, die die Ein–/Ausgabeschnittstelle angibt (Port–Adresse).

Beispiel: Angenommen, Sie geben den Befehl

```
I2F8
```

ein, und das Byte bei der Port–Adresse 02F8H habe den Wert FE dann zeigt DEBUG an:

```
FE
```

L (Lade Datei)

Format: **L** [***Adresse***[***Laufw:Satz Satz***]]

Bedeutung: Lädt eine vorher beim DEBUG–Aufruf angegebene oder mit dem **N**–Befehl (s. dort in diesem Kapitel) benannte Datei oder absolute Disk–Sektoren dieser Datei in den Speicher.

Adresse Eine gültige Adreßangabe, wie in Kapitel 11.4 beschrieben.

Laufw: Eine gültige Laufwerksangabe, wie in Kapitel 11.4 beschrieben, die das Laufwerk bezeichnet, von dem absolute Disksektoren geladen werden sollen.

Satz Zwei ganzzahlige Hexadezimalwerte, die den ersten Disk–Sektor sowie die absolute Anzahl zu ladender Disk–Sektoren bezeichnen. Maximal können 128 Sektoren (80H) geladen werden.

Wird der **L**–Befehl ohne Parameter oder nur mit ***Adresse*** benutzt, so wird die Datei, deren Spezifikation im Dateisteuerblock bei der Adresse CS:5C hinterlegt wurde, geladen. Ein korrekter Dateisteuerblock wird entweder durch die Angabe einer Datei beim Start von DEBUG oder durch den **N**–Befehl formatiert.

Beispiel: Angenommen, Sie haben nach dem Starten von DEBUG den Befehl

```
nprog.com
```

eingegeben. Um jetzt die Datei PROG.COM zu laden, brauchen Sie nur noch

```
l
```

einzugeben.

Wenn Sie dagegen nur Teile der Programmdatei PROG.COM laden wollen, können Sie z.B. folgendes eingeben:

```
L04BA:100 2 0F 6D
```

Jetzt lädt DEBUG 109 (6DH) Sektoren beginnend beim Sektor 15 (0FH) von der Festplatte (Laufwerk 2 = C) ab der Adresse 04BA:100 in den Speicher. Nach dem Laden meldet sich DEBUG wieder mit seiner Bereitschaftsanzeige (–).

Anmerkungen:

- DEBUG setzt beim **L**–Befehl die Register BX:CX auf die Anzahl gelesener Bytes.

- Wird der **L**–Befehl ganz ohne Parameter verwendet, wird die Datei ab Adresse CS:100 in den Speicher geladen.

- Hat die angegebene Datei die Namenserweiterung .EXE, wird sie im Speicher an die Adresse verschoben, die im Dateikopf der .EXE–Datei eingetragen ist. Bei .EXE–Dateien wird der Parameter ***Adresse*** im **L**–Befehl grundsätzlich ignoriert. Bevor die .EXE-Datei geladen wird, wird der Dateikopf entfernt. Deshalb unterscheidet sich die Größe einer geladenen .EXE–Datei von der im Disk–Verzeichnis eingetragenen.

- Wurde beim Aufruf von DEBUG oder beim **N**–Befehl eine Datei mit der Namenserweiterung .HEX angegeben, so wird davon ausgegangen, daß die Datei im Intel–Hexadezimalformat vorliegt und während des Ladens in eine ausführbare Form umgewandelt wird. In diesem Fall veranlaßt der **L**–Befehl ohne Parameter DEBUG, die Datei ab der Adresse zu laden, die in der HEX–Datei angegeben ist. Wird dagegen **L** mit einer Adresse benutzt, wird diese Adresse zu der in der HEX–Datei angegebenen addiert, um so die Lade–Startadresse zu ermitteln.

M (Verschiebe)

Format: **M** ***Bereich Adresse***

Bedeutung: Verschiebt den angegebenen Bereich in den Bereich, der bei der angegebenen Adresse beginnt.

Bereich Ein gültiger Adreßbereich im Format, wie in Kapitel 11.4 beschrieben, dessen Inhalt verschoben werden soll.

Adresse Eine gültige Adresse, bei der der Zielbereich, in den verschoben werden soll, beginnt.

Beispiel: Angenommen, Sie haben den Befehl

```
MCS:100 110 CS:500
```

eingegeben. DEBUG verschiebt dann zunächst das Byte bei CS:110 nach CS:510, danach das bei CS:10F nach CS:50F usw. Das Ergebnis kann mit dem **D**–Befehl (s. dort in diesem Kapitel) unter Angabe derselben Adressen angezeigt werden.

Anmerkungen:

- Sich überlappende Verschiebungen (das sind Verschiebungen, bei denen Teile des Quell– und Zielbereiches gemeinsame Adressen benutzen) werden immer ohne Datenverlust durchgeführt. Bytes, die überschrieben werden dürfen, werden deshalb immer zuerst verschoben.

- Bei Verschiebungen von höheren zu niedrigeren Adressen hin werden zunächst die Daten bei den niedrigeren Adressen des Blockes verschoben und dann zu den höheren Adressen hin weitergearbeitet.

- Bei Verschiebungen von niedrigeren zu höheren Adressen hin werden zunächst die Daten bei den höchsten Adressen des Blockes verschoben und dann zu den niedrigeren Adressen hin gearbeitet.

- Die Daten im Quellbereich bleiben unverändert, es sei denn, sie werden überschrieben.

N (Setze Name)

Format: **N** [*Laufw*:][*Pfad*]*Datei*[.*Erw*] [. . .]

Bedeutung: Der **N**–Befehl hat zwei Funktionen:

1. Er setzt eine vollständige Dateiangabe für nachfolgende **L**– oder **W**–Befehle, wenn DEBUG ohne Dateiangabe gestartet wurde.
2. Weist Dateinamen–Parameter der mit DEBUG bearbeiteten Datei zu. In diesem Fall akzeptiert der **N**–Befehl alle Parameter, die die bearbeitete Datei benutzt.

Beide Funktionen überlappen sich, wie diese Befehlsfolge verdeutlicht:

```
nprog.exe
l
g
```

Damit wird DEBUG zu folgenden Maßnahmen veranlaßt:

1. Der Dateiname PROG.EXE wird zunächst für nachfolgende **L**– oder **W**–Befehle gesetzt.
2. Der Dateiname PROG.EXE wird außerdem dem ersten Dateinamen–Parameter zugewiesen, der von jeder später mit DEBUG bearbeiteten Programmdatei benutzt wird.
3. Der **L**–Befehl lädt PROG.EXE in den Speicher.
4. Der **G**–Befehl startet das Programm, wobei PROG.EXE als einziger Dateinamenparameter benutzt wird, d.h. PROG.EXE wird so ausgeführt, als wäre `prog.exe` auf MS-DOS-Befehlsebene eingegeben worden.

***Laufw*:** Eine gültige Laufwerksangabe, wie in Kapitel 11.4 beschrieben. Wird ***Laufw*:** nicht angegeben, so nimmt DEBUG das aktuelle Laufwerk an.

Pfad Der Suchpfad für die angegebene Datei. Wird ***Pfad*** nicht angegeben, so nimmt DEBUG das aktuelle Verzeichnis der Disk im angegebenen Laufwerk an.

Datei.Erw Der Name für die zu ladende Datei.

Beispiele: Angenommen, Sie haben diese Befehlsfolge eingegeben:

```
nprog1.exe
l
ndatei1.dat datei2.dat
g
```

N setzt zunächst PROG1.EXE als Name für den anschließenden **L**–Befehl, mit dem die Datei PROG1.EXE in den Speicher geladen wird. Der **N**–Befehl wird dann erneut benutzt, um die Parameter, die von PROG1.EXE gebraucht werden, anzugeben. Wenn schließlich der **G**–Befehl gegeben wird, so wird das Programm PROG1.EXE genauso gestartet, als wenn auf der MS-DOS-Befehlsebene

```
prog1 datei1.dat datei2.dat
```

eingegeben worden wäre.

Beachten Sie: Wenn Sie jetzt einen **W**–Befehl geben würden, so würde die mit DEBUG bearbeitete Datei unter dem Namen DATEI1.DAT gespeichert. Um solche ungewollten Umbenennungen zu vermeiden, sollte vor jedem **L**– oder **W**–Befehl ein geeigneter **N**–Befehl benutzt werden.

Ein weiteres Beispiel zeigt eine typische Anwendung des **N**–Befehls:

```
debug prog.com
nparam1 param2/c
g
```

Hier wird durch **G** das Programm genauso gestartet, als wenn auf der MS–DOS–Befehlsebene

```
prog param1 param2/c
```

eingegeben worden wäre. Der Test und die Fehlersuche können dann also in einer normalen Laufzeit–Umgebung erfolgen.

Anmerkungen:

- Der **N**–Befehl kann vier Speicherbereiche beeinflussen:

  CS:5C Dateisteuerblock für Datei 1
  CS:6C Dateisteuerblock für Datei 2
  CS:80 Zeichenzähler
  CS:81 Gesamtzahl eingegebener Zeichen

  Für den ersten Dateinamen–Parameter, der beim **N**–Befehl angegeben wird, wird ein Dateisteuerblock bei der Adresse CS:5C angelegt. Wenn ein zweiter Dateinamen–Parameter angegeben wird, so wird für diesen der Dateisteuerblock bei der Adresse CS:6C angelegt. Die Gesamtzahl aller beim **N**–Befehl eingegeben Zeichen ohne das Befehlskürzel N selbst wird bei der Adresse CS:80 eingetragen. Ab der Adresse CS:81 wird schließlich die gesamte beim **N**–Befehl eingebene Parameter–Zeichenkette hinterlegt. In dieser Zeichenkette können auch Schalter und Trennzeichen enthalten sein, wie sie beim Programmaufruf auf der MS–DOS–Befehlsebene üblich sind.

O (Ausgabe über einen Port)

Format: **O** ***Wert Byte***

Bedeutung: Gibt ein Byte über die angegebene Ein–/Ausgabeschnittstelle aus.

Wert Eine gültige 16–Bit–Adresse, die die Ein–/Ausgabeschnittstelle angibt (Port–Adresse).

Byte Der auszugebende Wert im Bereich 00H bis FFH.

Beispiel: Angenommen, Sie wollen den Wert FEH über die Portadresse 02F8H ausgeben, dann müssen Sie folgenden Befehl eingeben:

```
o2f8 fe
```

Q (Beenden)

Format: **Q**

Bedeutung: Beendet die Arbeit mit DEBUG und kehrt auf die MS–DOS–Befehlsebene zurück, ohne die gerade bearbeitete Datei zu speichern.

R (Zeige Registerinhalte)

Format: **R** [***Registername***]

Bedeutung: Zeigt den Inhalt des angegebenen oder die Inhalte aller Register des Microprozessors (CPU).

Registername Einer der folgenden Registernamen:

AX	**BP**	**SS**
BX	**SI**	**CS**
CX	**DI**	**IP**
DX	**DS**	**PC**
SP	**ES**	**F**

IP und **PC** beziehen sich beide auf den Befehlszähler. Jede andere Angabe resultiert in der BR–Fehleranzeige (s. Kapitel 11.6).

Wird **F** als Registername angegeben, so werden alle CPU–Flags mit einem zwei Zeichen langen Namen, der gleichzeitig den Zustand der betreffenden Flag wiedergibt, angezeigt. Um eine der Flags zu ändern (setzen oder löschen), muß einer der folgenden Setz– oder Löschcodes bei der Bereitschaftsanzeige eingegeben werden:

Flag–Name	setzen	löschen
Überlauf	**OV** (ja)	**NV** (nein)
Richtung	**DN** (Dekrement)	**UP** (Inkrement)
Interrupt	**EI** (erlaubt)	**DI** (verboten)
Vorzeichen	**NG** (negativ)	**PL** (positiv)
Null	**ZR** (ja)	**NZ** (nein)
Zusätzl. Übertrag	**AC** (ja)	**NA** (nein
Parität	**PE** (gerade)	**PO** (ungerade)
Übertrag	**CY** (ja)	**NC** (nein)

In Folge des **RF**–Befehls können mehrere Flag–Setz– oder –Löschkennungen eingegeben werden. Trennzeichen sind nicht erforderlich. Alle nicht angegebenen Flags bleiben unverändert.

Beispiele: Wenn Sie den Befehl

```
r
```

z.B. bei der aktuellen Speicheradresse CS:11A eingeben, erhalten Sie eine Anzeige der folgenden Art:

```
AX=0E00   BX=00FF  CX=0007  DX=01FF  SP=039D  BP=0000
SI=005C   DI=0000  DS=04BA  ES=04BA  SS=04BA  CS=04BA
IP=011A   NV UP DI NG NZ AC PE NC
04BA:011A CD21     INT      21                DS=04BA
```

Die Zeilenaufteilung ist auf dem Bildschirm etwas anders.

Wenn Sie dann den RF–Befehl benutzen, erhalten Sie z.B.:

```
rf
NV UP DI NG NZ AC PE NC - _
```

Jetzt können Sie bei der Cursor–Position jede gültige Flag–Kennung eingeben, also z.B:

```
NV UP DI NG NZ AC PE NC - PLEICY
```

Wenn Sie zum Abschluß die Eingabetaste drücken, meldet sich DEBUG in der Folgezeile wieder mit seiner Bereitschaftsanzeige. Um die neuen Flag–Zustände zu sehen, können Sie wieder den **RF**–Befehl verwenden, also:

```
rf
NV UB EI PL NZ AC PE CY - _
```

Drücken der Eingabetaste beendet den **R**–Befehl.

Anmerkungen:

- Wird für eine Flag mehr als ein Setz–oder Löschcode eingegeben, zeigt DEBUG den DF–Fehler (s. Kapitel 11.6). Wird ein ungültiger Setz– oder Löschcode eingegeben, zeigt DEBUG den BF–Fehler. In beiden Fällen werden alle ggf. in der Liste angegebenen Flags bis zur falsch definierten geändert; die falsch definierte sowie alle ggf. noch folgenden nicht.

- Nach dem Start von DEBUG werden die Segmentregister der CPU auf die niedrigste Adresse des freien Speichers, der Befehlszähler auf 0100H gesetzt, alle Flags gelöscht und alle übrigen Register auf Null gesetzt.

S (Suche)

Format: **S *Bereich Liste***

Bedeutung: Durchsucht den angegebenen Bereich nach der angegebenen Liste von Byte–Werten.

Bereich Ein gültiger Adreßbereich im Format, wie in Kapitel 11.4 beschrieben, der durchsucht werden soll.

Liste Eine Liste von Bytes, wie in Kapitel 11.4 beschrieben, nach der im angegebenen Bereich gesucht werden soll.

DEBUG zeigt alle Adressen an, bei der die angegebene Liste gefunden wird, jede für sich in einer Zeile.

Beispiel: Angenommen, Sie haben den folgenden Befehl eingegeben:

```
scs:100 200 41
```

Dann erhalten Sie beispielsweise die Anzeige:

```
28EE:019F
28EE:01B6
```

Anmerkungen:

- Die angegebene Liste kann aus einem oder mehreren Bytes, getrennt durch Leerstellen oder Kommata (,), bestehen. Besteht sie nur aus einem Byte, wird nur die Adresse des ersten Bytes in der Liste angezeigt.

T (Verfolge Programmablauf)

Format: **T** [=***Adresse***][***Wert***]

Bedeutung: Führt einen Befehl aus und zeigt dann alle Registerinhalte, alle Flag–Zustände sowie den decodierten Befehl an.

Adresse Eine gültige Adresse, wie in Kapitel 11.4 beschrieben, bei der die Programmablaufverfolgung beginnt. Wird ***Adresse*** nicht angegeben, so wird der Befehl, auf den der aktuelle Befehlszähler zeigt, ausgeführt.

Wert Eine ganze Zahl, die die Anzahl der auszuführenden Befehle angibt. Nach jedem ausgeführten Befehl werden die Registerinhalte und Flag–Zustände angezeigt. Wird ***Wert*** nicht angegeben, so wird nur ein Befehl ausgeführt.

Beispiele: Wenn Sie den Befehl

```
t
```

z.B. bei der aktuellen Speicheradresse 04BA:011A eingeben, erhalten Sie eine Anzeige der folgenden Art:

```
AX=0E00   BX=00FF  CX=0007  DX=01FF  SP=039D  BP=0000
SI=005C   DI=0000  DS=04BA  ES=04BA  SS=04BA  CS=04BA
IP=011A   NV UP DI NG NZ AC PE NC
04BA:011A CD21     INT      21                DS=04BA
```

Die Zeilenaufteilung ist auf dem Bildschirm etwas anders.

Geben Sie dagegen

```
t=011a 10
```

ein, so werden 16 Befehle (10H), beginnend bei der Adresse 011AH im aktuellen Segment, ausgeführt und nach jedem ausgeführten Befehl werden, wie oben be-

schrieben, alle Registerinhalte und Flag–Zustände angezeigt. Um den Ablauf verfolgen zu können, können Sie die Anzeige mit Ctrl–NumLock oder Ctrl–S anhalten und mit jeder anderen Taste fortsetzen.

Anmerkungen:

- Der **T**–Befehl benutzt den hardwaremäßig vorgesehenen Programmablauf–Verfolgungsmodus des 8086/8088–Microprozessors. Sie können deshalb auch die Abarbeitung von Programmteilen im ROM (Festwertspeicher) mit dem **T**–Befehl verfolgen.

U (Disassembliere)

Format: **U** [***Bereich***]

Bedeutung: Disassembliert den Maschinensprachecode im angegebenen Bereich und zeigt ihn in Form einer Assemblerliste auf dem Bildschirm an.

Bereich Ein Adreßbereich, wie in Kapitel 11.4 beschrieben. Wird ***Bereich*** nicht angegeben, so werden die ersten 32 Bytes, beginnend bei der Adresse vom letzten **U**–Befehl disassembliert. Wurde vorher noch kein **U**–Befehl benutzt, beginnt die Disassemblierung bei Adresse CS:0100. Wird für Bereich nur eine Adresse angegeben. so werden die nächsten 32 Bytes, die bei dieser Adresse beginnen, disassembliert.

Beispiel: Wenn Sie eingeben:

```
U04BA:100 L10
```

erhalten Sie z.B.:

```
04BA:0100 A1E602   MOV   AX,[02E6]
04BA:0103 80FC10   CMP   AH,10
04BA:0106 7403     JZ    010B
04BA:0108 E8CA01   CALL  02D5
04BA:010B F9       STC
04BA:010C C3       RET
04BA:010D E87C47   CALL  488C
```

Wenn Sie dagegen eingeben:

```
U04BA:0100 0108
```

erhalten Sie:

```
04BA:0100 A1E602   MOV   AX,[02E6]
04BA:0103 80FC10   CMP   AH,10
04BA:0106 7403     JZ    010B
04BA:0108 E8CA01   CALL  02D5
```

Anmerkungen:

- Manchmal kann die Anzahl disassemblierter Bytes größer als die angegebene Anzahl sein, da die Assemblerbefehle unterschiedlich lang sind, und auf jeden Fall immer ein ganze Befehl disassembliert wird.

- Enthält der angegebene Bereich keinen gültigen Maschinensprachecode, so wird die Anzeige fehlerhaft.

W (Schreibe auf Disk)

Format: **W** [***Adresse***[***Laufw:Satz Satz***]]

Bedeutung: Schreibt den angegebenen Speicherbereich auf Disk.

Adresse Eine gültige Adresse, wie in Kapitel 11.4 beschrieben. Wird ***Adresse*** nicht angegeben, so werden die Daten beginnend bei Adresse CS:0100 gespeichert.

Laufw: Eine gültige Laufwerksangabe, die das Laufwerk bezeichnet, auf dessen Disk die Daten geschrieben werden sollen.

Satz Zwei ganzzahlige Hexadezimalwerte, die den ersten logischen Disksektor sowie die Anzahl zu schreibender Disksektoren bezeichnen. Maximal können 128 Sektoren (80H) gespeichert werden.

Wird der **W**–Befehl ohne Parameter benutzt, so muß das Registerpaar BX:CX auf die Anzahl zu schreibender Bytes gesetzt werden. Außerdem werden die Daten in die Datei geschrieben, deren Spezifikation im Dateisteuerblock bei der Adresse CS:5C hinterlegt wurde. Ein korrekter Dateisteuerblock wird bei dieser Adresse entweder durch die Angabe einer Datei beim Start von DEBUG oder durch den **N**–Befehl (s. dort in diesem Kapitel) hinterlegt.

Beispiele: Angenommen, Sie haben DEBUG mit dem Befehl

```
debug prog.com
```

gestartet. Um jetzt die im Speicher bearbeitete Datei zurückzuspeichern, brauchen Sie nur noch

```
w
```

einzugeben. Beachten Sie hier aber, daß Sie das Registerpaar BX:CX wiederherstellen müssen, wenn Sie vorher den **G**– oder **T**–Befehl benutzt haben.

Wenn Sie dagegen nur Teile der bearbeiteten Datei zurückschreiben wollen, können Sie eingeben:

```
w04BA:100 2 0F 6D
```

Jetzt schreibt DEBUG 109 (6DH) Sektoren, beginnend beim Sektor 15 (0FH) auf die Festplatte (Laufwerk 2 = C). Nach dem Schreiben meldet sich DEBUG wieder mit seiner Bereitschaftsanzeige (–).

Anmerkungen:

- Beim Speichern von Daten in Sektoren muß äußerst vorsichtig vorgegangen werden, weil durch eine falsche Sektorangabe sämtliche Daten in diesem Bereich auf der Disk zerstört werden.

- Wird nur ein Adreßoffset angegeben, so nimmt DEBUG das CS–Segment an.

- Wurde DEBUG mit einer Dateiangabe gestartet und wurden dann **N**–Befehle benutzt, so muß ggf. vor dem Speichern mit einem erneuten **N**–Befehl wieder der richtige Name und damit Dateisteuerblock gesetzt werden.

- Wird versucht, in einen Datei mit der Namenserweiterung .EXE oder .HEX zu schreiben, so zeigt DEBUG eine Fehlermeldung an, da diese Dateiformate von DEBUG nicht erzeugt werden können. Muß eine solche Datei geändert werden, so muß Sie vor der Benutzung von DEBUG umbenannt werden. Danach kann sie mit dem DEBUG–Befehl **L** geladen und im Speicher bearbeitet werden. **T**– oder **G**–Befehle dürfenn hier jedoch **nicht** benutzt werden. Anschließend kann sie mit **W** zurückgespeichert und schließlich wieder umbenannt werden.

11.6 Fehlermeldungen bei DEBUG

Während der Arbeit mit DEBUG können die unten angegebenen Fehlermeldungen erscheinen. Jeder dieser Fehlerfälle beendet den gegenwärtig ausgeführten DEBUG–Befehl, **nicht** jedoch das DEBUG–Programm selbst.

Fehlercode	Bedeutung
BF	Falsche Flag. Sie haben versucht, einen Flagzustand zu ändern, haben jedoch nicht die richtige Buchstabenkombination dafür angegeben. Beim **R**–Befehl in Kapitel 11.5 sind die erlaubten Flag–Bezeichnungen aufgeführt.
BP	Zu viele Unterbrechungspunkte. Sie haben beim **G**–Befehl mehr als 10 Unterbrechungsadressen angegeben. Wiederholen Sie den **G**–Befehl mit maximal 10 Unterbrechungsadressen.
BR	Falsches Register. Sie haben im **R**–Befehl einen falschen Registernamen benutzt. Bei der Beschreibung des **R**–Befehls sind die erlaubten Registernamen angegeben.
DF	Doppelte Flag–Angabe. Sie haben für eine Flag–Änderung mit dem **R**–Befehl zwei Änderungscodes eingegeben. Es ist nur einer erlaubt.

Anhang A: Meldungen und Fehlermeldungen

A.1 Disk– und Gerätefehlermeldungen

In diesem Abschnitt sind alle Fehlermeldungen alphabetisch aufgeführt, die im Zusammenhang mit Peripheriegeräten wie Tastatur, Disks, Druckern usw. vorkommen können.

Wenn ein solcher Fehler auftritt, meldet MS–DOS diesen in folgendem allgemeinen Format:

```
Typ Aktion Gerät
Abort,Retry,Ignore:_
```

(Abbrechen,Wiederholen,Ignorieren). Je nachdem, welche Maßnahme gewünscht wird, kann mit A (Abort=Abbrechen), R (Retry=Wiederholen) oder I (Ignore=Ignorieren) geantwortet werden. Die Typ–, Aktions– und Gerätefehlermeldungen werden im folgenden in getrennten Abschnitten aufgeführt.

Typ–Fehlermeldungen

Bad call format error (Falscher Aufruf)
Einem Gerätetreiber wurde ein Kopfsatz falscher Länge übergeben.

Bad command error (Falscher Befehl)
Ein Gerätetreiber hat an das angegebene Gerät einen falschen Befehl ausgegeben.

Bad unit error (Falsches Gerät)
Einem Gerätetreiber wurde eine falsche Untergerätenummer übergeben.

Data error (Datenfehler)
MS–DOS kann die Daten von der Disk nicht sicher lesen. Dieser Fehler wird häufig von beschädigten Disketten erzeugt. Versuchen Sie durch mehrmalige Eingabe von R (Wiederholen) die Daten doch noch zu lesen oder brechen Sie mit A (Abbrechen) das Programm ab. Kopieren Sie die Disk ggf., um noch lesbare Daten zu retten.

FCB unavailable (FCB nicht verfügbar)
Bei aktiviertem gemeinsamem Dateizugriff (Netzwerkbetrieb) wurde versucht, gleichzeitig mehr Dateien zu öffnen, als im FCBS–Befehl der CONFIG.SYS–Datei (s. Kapitel 2.3) angegeben wurde.

General Failure error (Allgemeiner Fehler)
Ein nicht beschriebener Fehler ist aufgetreten. Kann durch die Eingabe von R (Wiederholen) der Fehler nicht beseitigt werden, so muß ein erfahrener Programmierer zu Rate gezogen werden.

Invalid disk change error (Nicht erlaubter Diskettenwechsel)
Sie haben in einem Laufwerk eine Diskette gewechselt, obwohl das Programm Sie nicht dazu aufgefordert hat. Legen Sie die ursprüngliche Diskette wieder ein und geben Sie R (Wiederholen) ein.

Lock violation error (Dateizugriff nicht möglich)
Sie haben versucht, auf einen Dateibereich zuzugreifen, der gerade von einem anderen Programm benutzt wird. Drücken Sie A für Abbruch oder warten Sie etwas und drücken Sie dann R (Wiederholen).

No paper error (Kein Papier im Drucker)
Der Drucker hat entweder kein Papier oder ist ausgeschaltet . Legen Sie Papier ein oder schalten Sie den Drucker ein und drücken Sie R (Wiederholen).

Non–DOS disk error (Keine DOS–Diskette)
MS–DOS kann das Diskformat der bezogenen Diskette nicht interpretieren, weil Informationen fehlen oder die Diskette ein anderes Betriebssystem enthält. Versuchen Sie mit CHKDSK die Diskette zu analysieren. Falls das nicht zum Erfolg führt, formatieren Sie die Diskette mit dem FORMAT–Befehl neu (die Daten werden dabei allerdings zerstört).

Not ready error (Nicht bereit)
Das angegebene Gerät (meistens ein Laufwerk oder Drucker) ist für die Datenübertragung nicht bereit. Diskette einlegen und Laufwerk schließen oder Drucker auf On Line schalten.

Read fault error (Lesefehler)
MS–DOS kann vom angegebenen Gerät (meistens ein Disk–Laufwerk) keine Daten lesen. Überprüfen Sie, ob die Diskette richtig zentriert und eingelegt ist und antworten Sie dann mit R (Wiederholen).

Sector not found error (Sektor nicht gefunden)
Die bezogene Disk hat eine fehlerhafte Stelle, so daß MS–DOS die angeforderte Information nicht findet. Kopieren Sie alle kompletten Dateien auf eine andere Disk und versuchen Sie die defekte Disk neu zu formatieren.

Seek error (Positionierungsfehler)
MS–DOS kann die richtige Spur auf der Disk im bezogenen Laufwerk nicht finden. Stellen Sie sicher, daß die Diskette richtig eingelegt ist und versuchen Sie mit R zu wiederholen.

Sharing violation (Dateizugriff nicht möglich)
Sie haben versucht, auf einen Dateibereich zuzugreifen, der gerade von einem anderen Programm benutzt wird. Drücken Sie A für Abbruch oder warten Sie etwas und drücken Sie dann R (Wiederholen).

Write fault error (Schreibfehler)
MS–DOS kann auf das angegebene Gerät (meistens ein Disk–Laufwerk) keine Daten schreiben. Überprüfen Sie, ob die Diskette richtig zentriert und eingelegt ist und antworten Sie dann mit R (Wiederholen).

Write protect error (Schreibschutz)
Sie haben versucht, auf eine schreibgeschützte Disk zu schreiben. Vergewissern Sie sich, ob Sie wirklich auf diese Disk schreiben wollen. Entfernen Sie dann ggf. die Schreibschutzlasche von der Schreibschutzkerbe und geben Sie R für Wiederholen ein.

Aktions–Meldungen

Die Aktionsmeldungen zeigen an, bei welcher Datenübertragung der Fehler aufgetreten ist. Es gibt nur zwei, nämlich

reading (Lesen)
writing (Schreiben)

Geräte–Meldungen

Unter *Gerät* im allgemeinen Fehlermeldungsformat (s. Anfang dieses Kapitels) wird der Name des den Fehler verursachenden Gerätes angezeigt, also z.B

PRN (Drucker)
drive A (Laufwerk A)

MS–DOS wartet dann, bis Sie eine der folgenden Antworten eingeben:

A Abort = Abbrechen. Beendet das Programm, das die Lese– oder Schreibanforderung gegeben hat.

R Retry = Wiederholen. Wiederholt die Operation. Diese Antwort sollte im Allgemeinen gegeben werden, wenn die Fehlerursache behoben wurde.

I Ignore = Ignorieren. Ignoriert die Fehlerursache (z.B. einen fehlerhaften Sektor) und fährt mit der Verarbeitung vort. Dies kann zu Datenverlust führen.

Normalerweise wird zur Fehlerbeseitigung zunächst mit **R** versucht, mit dem Programm fortzufahren. Gelingt dies nicht, wird mit **A** beendet.

Es gibt noch eine weitere Fehlermeldung, die auf fehlerhaftes Lesen oder Schreiben bei Disk–Zugriffen hindeutet:

File Allocation Table bad for drive x

(Dateizuordnungstabelle auf Laufwerk *x* defekt). Diese Meldung bedeutet, daß die Kopie der Dateizuordnungstabelle eines bestimmten Laufwerkes, die sich MS–DOS beim ersten Zugriff in den Speicher lädt, Verweise auf nichtexistente Blöcke enthält. Möglicherweise wurde die benutzte Disk vorher gar nicht oder nicht korrekt formatiert. Wenn dieser Fehler nach der Formatierung wieder auftritt, ist die Disk defekt und sollte nicht mehr verwendet werden.

A.2 Alphabetisches Verzeichnis aller MS–DOS–Meldungen

Dieser Abschnitt beschreibt alle MS–DOS–Meldungen, soweit sie nicht gerätespezifisch (s. Kapitel A.1) sind. Die Beschreibungen enthalten die Ursache und im Fehlerfall Hinweise auf die Fehlerbeseitigung. Die MS–DOS–Befehle oder –Programme, bei deren Benutzung die jeweilige Fehlermeldung angezeigt wurde, werden in eckigen Klammern ([ ]) angegeben.

Abort edit (Y/N)? (Edierung beenden (J/N)?)
[EDLIN]
MS–DOS zeigt diese Meldung an, wenn Sie bei EDLIN den Q–Befehl zum

Beenden gegeben haben. Beim Q–Befehl wird EDLIN beendet, ohne den edierten Text zu speichern. Geben Sie Y für Ja oder N für Nein ein.

Abort,Retry,Ignore? (Abbrechen,Wiederholen,Ignorieren?)
[MS–DOS]
Diese Anzeige erscheint immer zusammen mit einer Gerätefehlermeldung (s. Kapitel A.1).

Access denied (Zugriff abgelehnt)
[ATTRIB][FIND][PRINT][REPLACE][XCOPY]
Sie haben versucht, eine schreibgeschützte oder gegen gegenseitigen Dateizugriff verriegelte Datei zu ersetzen.

Add *Dateiname*? (Y/N) (Datei *Dateiname* hinzufügen? (J/N))
[REPLACE]
Wenn Sie beim REPLACE–Befehle den /W–Schalter angegeben haben, wartet das Programm auf Bestätigung. Bei Y (Yes=Ja) wird die Datei hinzugefügt, bei N (No=Nein) nicht.

Adding *Dateiname* (Datei *Dateiname* wird hinzugefügt)
[REPLACE]
Der REPLACE–Befehl zeigt diese Meldung an, wenn die angegebene Datei zum Zielverzeichnis hinzugefügt wird.

All files canceled by operator (Alle Dateien vom Bediener gelöscht)
[PRINT]
der PRINT–Befehl zeigt diese Meldung an, wenn dort der /T–Schalter benutzt wurde.

Allocation error, size adjusted (Zuweisungsfehler, Größe korrigiert)
[CHKDSK]
Die im Verzeichnis angegebene Datei hat eine andere Größe, als ihr in der Dateizuordnungstabelle zugewiesen wird. Die Dateigröße wird von CHKDSK dem Tabelleneintrag angepaßt.

All specified file(s) are contiguous (Alle angegebenen Dateien sind zusammenhängend)
[CHKDSK]
Diese Meldung zeigt CHKDSK an, wenn es keine Lücken in den Dateien bei der Disküberprüfung gefunden hat.

APPEND/ASSING Conflict (APPEND/ASSIGN–Konflikt)
[APPEND]
Sie haben verbotenerweise versucht, den APPEND–Befehl bei einem Laufwerk anzuwenden, das bereits mit dem ASSIGN–Befehl umdefiniert wurde. Löschen Sie die Umdefinition für dieses Laufwerk, ehe sie den APPEND–Befehl darauf anwenden.

Are you sure (Y/N)? (Sind Sie sicher (J/N)?)
[MS–DOS]
MS–DOS fordert mit dieser Anzeige die Bestätigung, daß alle Dateien des aktuellen Verzeichnisses mit dem Befehl DEL *.* gelöscht werden sollen. Bei Y (Yes = Ja) wird der Befehl ausgeführt, bei N (No = Nein) nicht.

Attempted write-protect violation (Schreibschutz – Schreiben nicht möglich)
[FORMAT]
Sie haben versucht, eine schreibgeschützte Diskette zu formatieren. Überlegen Sie, ob diese Diskette formatiert werden darf und entfernen Sie ggf. die aufgeklebte Schreibschutzlasche.

Backing up files to drive *x:**
Diskette Number: *n* (***Dateien werden auf Laufwerk *x:* gesichert***
Diskette Nummer: *n*)
[BACKUP]
Der BACKUP–Befehl zeigt diese informative Meldung an, während Dateien auf die angegebene Zieldisk gesichert werden. Vergessen Sie nicht, die Sicherungsdisketten auch mit der zugehörigen Nummer zu beschriften, damit ein ggf. notwendiges Rückspeichern mit RESTORE erleichtert wird.

Bad call format reading/writing drive *x:* (Falscher Aufruf)
[MS–DOS]
Dieses ist eine Geräte–Fehlermeldung (s. Kapitel A.1).

Bad command error reading/writing drive *x:* (Falscher Befehl)
[MS–DOS]
Dieses ist eine Geräte–Fehlermeldung (s. Kapitel A.1).

Bad command or file name (Falscher Befehl oder Dateiname)
[MS–DOS]
Der Befehlsprozessor COMMAND.COM kann den eingegebenen Befehl oder das Programm nicht finden. Stellen Sie sicher, daß der Befehl korrekt eingegeben wurde oder daß das Programm auch auf der angegebenen Diskette vorhanden ist.

Bad or missing Command Interpreter (Falscher oder fehlender Befehlsprozessor)
[MS–DOS]
MS–DOS kann die COMMAND.COM–Datei nicht auf der Disk im angegebenen Laufwerk finden. Entweder ist die Datei nicht im Basisverzeichnis enthalten oder Sie ist ungültig. Starten Sie entweder MS–DOS mit einer Disk, die COMMAND.COM enthält oder kopieren Sie COMMAND.COM von der Original–Systemdiskette auf Ihre Arbeitsdisk.

Bad or missing *Dateiname* (Falscher oder fehlender *Dateiname*)
[MS–DOS]
MS–DOS kann einen in der CONFIG.SYS–Datei angegebenen Gerätetreiber nicht finden. Überprüfen Sie alle in der CONFIG.SYS–Datei angegebenen DEVICE–Befehle.

Bad Partition Table (Falsche Partitionstabelle)
[FORMAT]
Die Festplatte enthält keine Partitionstabelle. Sie müssen zuerst mit FDISK (s. Kapitel 3.2.1) eine MS–DOS–Partition auf Ihrer Festplatte einrichten.

Bad unit error reading drive *x:* (Falsches Gerät)
[MS–DOS]
Dieses ist eine Geräte–Fehlermeldung (s. Kapitel A.1).

BREAK is off/on (BREAK ist AUS/AN)
[MS–DOS]
Diese informative Meldung zeigt an, ob die Überwachung der Ctrl–Break-Funktion bei Schreib–/Leseoperationen ein– oder ausgeschaltet ist, wenn BREAK ohne Parameter eingegeben wird.

Cannot CHDIR to path –
tree past this point not processed (Pfad kann nicht gefunden werden. Baumstruktur unterhalb von diesem Punkt wird nicht verarbeitet)
[CHKDSK]
Während der Überprüfung der Verzeichnisstruktur kann CHKDSK nicht zum angegebenen Verzeichnis verzweigen. Alle ggf. in diesem Verzeichnis enthaltenen Unterverzeichnisse werden nicht mehr verifiziert.

Cannot CHDIR to root (Verzweigung zum Basisverzeichnis nicht möglich)
[CHKDSK]
Während der Überprüfung der Verzeichnisstruktur kann CHKDSK nicht zurück ins Basisverzeichnis verzweigen und kann deshalb ggf. weitere

vorhandene Unterverzeichnisse nicht mehr überprüfen. Falls dieser Fehler nach erneutem Starten von MS–DOS wieder angezeigt wird, ist die bezogene Disk unbrauchbar.

Cannot CHKDSK a Network Drive (Netzwerk–Laufwerk mit CHKDSK nicht überprüfbar)
[CHKDSK]
Disk in innerhalb eines Netzwerkes zugewiesenen Laufwerken können nicht mit CHKDSK überprüft werden.

Cannot CHKDSK a SUBSTed or ASSIGNed drive (Ein mit SUBST oder ASSIGN zugewiesenes Laufwerk nicht mit CHKDSK überprüfbar)
[CHKDSK]
Disks in substituierten oder zugewiesenen Laufwerken können mit CHKDSK nicht überprüft werden.

Cannot COPY from/to a reserved device (COPY von einem oder auf ein reserviertes Gerät nicht möglich)
[XCOPY]
Sie können von einem oder auf ein Gerät nicht kopieren.

Cannot create Subdirectory BACKUP on drive *x:* (Unterverzeichnis BACKUP kann nicht auf Laufwerk *x:* eingerichtet werden)
[BACKUP]
BACKUP kann sein Unterverzeichnis für die gesicherten Dateien nicht erzeugen, weil die Disk schreibgeschützt oder voll ist oder ein Unterverzeichnis BACKUP bereits existiert und dieses schreibgeschützt ist.

Cannot DISKCOMP to or from an ASSIGNed or SUBSTed drive (DISKCOMP bei einem mit SUBST oder ASSIGN zugewiesenen Laufwerk nicht möglich)
[DISKCOMP]
Sie haben versucht, DISKCOMP auf ein substituiertes oder zugewiesenes Laufwerk anzuwenden.

Cannot DISKCOMP to or from a network drive (DISKCOMP bei einem Netzwerk–Laufwerk nicht möglich)
[DISKCOMP]
Sie haben versucht, DISKCOMP auf ein in einem Netzwerk zugewiesenes Laufwerk anzuwenden.

Cannot DISKCOPY to or from an ASSIGNed or SUBSTed drive (DISKCOPY bei einem mit SUBST oder ASSIGN zugewiesenen Laufwerk nicht möglich)
[DISKCOPY]
Sie haben versucht, DISKCOPY auf ein substituiertes oder zugewiesenes Laufwerk anzuwenden.

Cannot DISKCOPY to or from a network drive (DISKCOPY bei einem Netzwerk–Laufwerk nicht möglich)
[DISKCOPY]
Sie haben versucht, DISKCOPY auf ein in einem Netzwerk zugewiesenes Laufwerk anzuwenden.

Cannot do binary reads from a device (Binäres Lesen von einem Gerät nicht möglich)
[COPY]
Beim Kopieren von einem Gerät ist der Binär–Modus nicht möglich. Verwenden Sie entweder nicht den /B–Schalter beim COPY–Befehl oder geben Sie den /A–Schalter für Kopieren im ASCII–Modus an.

Cannot edit .BAK file--rename file (Edieren einer .BAK–Datei nicht möglich--Datei umbenennen)
[EDLIN]
Sie haben versucht, eine Datei mit der Namenserweiterung .BAK zu edieren, also eine EDLIN–Sicherungskopie. Erstellen Sie entweder eine Kopie mit anderem Namen oder benennen Sie die Datei um.

Cannot FORMAT an ASSIGNed or SUBSTed drive (FORMAT bei einem mit SUBST oder ASSIGN zugewiesenen Laufwerk nicht möglich)
[FORMAT]
Sie haben versucht, FORMAT auf ein substituiertes oder zugewiesenes Laufwerk anzuwenden.

Cannot FORMAT a network drive (FORMAT bei einem Netzwerk-Laufwerk nicht möglich)
[FORMAT]
Sie haben versucht, FORMAT auf ein in einem Netzwerk zugewiesenes Laufwerk anzuwenden.

Cannot JOIN a network drive (JOIN bei einem Netzwerk–Laufwerk nicht möglich)
[JOIN]

Sie haben versucht, JOIN auf ein in einem Netzwerk zugewiesenes Laufwerk anzuwenden.

Cannot LABEL a network drive (LABEL bei einem Netzwerk–Laufwerk nicht möglich)
[LABEL]
Sie haben versucht, LABEL auf ein in einem Netzwerk zugewiesenes Laufwerk anzuwenden.

Cannot LABEL an ASSIGNed or SUBSTed drive (LABEL bei einem mit SUBST oder ASSIGN zugewiesenen Laufwerk nicht möglich)
[LABEL]
Sie haben versucht, LABEL auf ein substituiertes oder zugewiesenes Laufwerk anzuwenden.

Cannot perform a cyclic copy (Zyklisches Kopieren nicht möglich)
[XCOPY]
Wenn Sie beim XCOPY–Befehl den /S–Schalter benutzen, dürfen Sie als Zielverzeichnis kein Unterverzeichnis der Quelle angeben.

Cannot recover . entry, processing continued (.–Verzeichnis nicht wiederherstellbar, Verarbeitung wird fortgesetzt)
[CHKDSK]
Bei der Überprüfung der Verzeichnisstruktur einer Disk wurde ein fehlerhafter .–Eintrag (aktuelles Verzeichnis) gefunden, der nicht korrigierbar ist. CHKDSK fährt mit der Überprüfung dennoch fort.

Cannot RECOVER a network drive (RECOVER bei einem Netzwerk–Laufwerk nicht möglich)
[RECOVER]
Sie haben versucht, RECOVER auf ein in einem Netzwerk zugewiesenes Laufwerk anzuwenden.

Cannot SUBST a network drive (SUBST bei einem Netzwerk–Laufwerk nicht möglich)
[SUBST]
Sie haben versucht, SUBST auf ein in einem Netzwerk zugewiesenes Laufwerk anzuwenden.

Cannot SYS to a network drive (SYS bei einem Netzwerk–Laufwerk nicht möglich)
[SYS]

Sie haben versucht, die Systemdateien auf eine Disk in einem Laufwerk, das in einem Netzwerk zugewiesen wurde, zu übertragen.

Cannot use PRINT – Use NET PRINT (PRINT nicht erlaubt – NET PRINT verwenden)
[PRINT]
Sie haben versucht, beim Netzwerkbetrieb mit PRINT eine Datei auszudrukken. Verwenden Sie stattdessen den NET PRINT–Befehl (siehe dazu im Handbuch *Microsoft Networks User's Guide*).

CHDIR . . failed, trying alternate method (CHDIR . . fehlerhaft, alternative Methode wird versucht)
[CHKDSK]
Bei der Überprüfung der Verzeichnisstruktur kann CHKDSK nicht zu einem Vaterverzeichnis zurückkehren. Es wird versucht, in dieses Verzeichnis über das Basiverzeichnis zu gelangen.

Compare another diskette (Y/N)? (Noch eine Diskette vergleichen (J/N)?)
[DISKCOMP]
Diese Anfrage erscheint am Ende eines Disk–Vergleiches mit DISKCOMP. Bei Y (Yes = Ja) werden weitere Disketten verglichen, bei N (No = Nein) wird beendet.

Compare error on disk
side *s*, track *t* (Vergleichsfehler auf Disk; Seite *s*, Spur *t*)
[DISKCOMP]
Beim Diskettenvergleich meldet DISKCOMP einen Unterschied auf Seite *s* bei der Spur *t*.

Compare OK (Vergleich in Ordnung)
[DISKCOMP]
Diese informative Meldung zeigt DISKCOMP an, wenn beide verglichenen Disks identisch sind.

Compare process ended (Vergleich abgebrochen)
[DISKCOMP]
Beim Diskvergleich mit DISKCOMP ist ein fataler Fehler aufgetreten, der eine Fortsetzung des Vergleichs unmöglich macht.

Comparing *t* tracks
***n* sectors per track, *s* side(s)** (Vergleich mit *t* Spuren, *n* Sektoren/Spur, *s* Seite(n))

[DISKCOMP]
Mit dieser Meldung zeigt DISKCOMP das Format, in dem beide zu vergleichenden Disks formatiert wurden.

COM port does not exist (Datenfernübertragungsschnittstelle existiert nicht)
[MODE]
Sie haben beim MODE–Befehl eine ungültige Nummer für die serielle Schnittstelle eingegeben. Erlaubt sind nur COM1: und COM2:.

Contains *n* non-contiguous blocks (Enthält *n* nicht zusammenhängende Blöcke)
[CHKDSK]
Die mit CHKDSK überprüfte Disk enthält fragmentierte Dateien. Wenn Sie eine solche Disk kopieren wollen, sollten Sie statt DISKCOPY die Befehle COPY oder XCOPY verwenden, da diese die Dateien in sequentiell zusammengehörigen Sektoren im Ziel ablegen.

Content of destination lost before copy (Inhalt der Zieldatei vor dem Kopieren verloren)
[COPY]
Die beim COPY–Befehl angegebene Quelldatei wurde vor dem Ende des Kopiervorganges überschrieben. Sehen Sie bei der Beschreibung des COPY–Befehls im Kapitel 6.3.1 nach der korrekten Syntax.

Convert lost chains to files (Y/N)? (Verlorenen Blöcke in Dateien umwandeln (J/N)?)
[CHKDSK]
CHKDSK hat bei der Disküberprüfung Blöcke gefunden, die zwar zugewiesen sind, aber nicht mit einer der Dateien verknüpft werden konnten. Wird mit Y (Yes = Ja) geantwortet, so wird für jede Zeichenkette ein Dateieintrag der Form FILE*nnnn*.CHK erstellt, beginnend mit *nnnn* = 0001. Wird mit N (No = Nein) geantwortet, so werden diese Blöcke freigegeben und die darin enthaltenen Daten sind verloren.

Copy another Diskette (Y/N)? (Noch eine Diskette kopieren (J/N)?)
[DISKCOPY]
Nach der Duplizierung einer Diskette mit DISKCOPY wird angefragt, ob weiter dupliziert werden soll. Bei Y (Yes = Ja) wird DISKCOPY neu gestartet, bei N (No = Nein) kehrt DISKCOPY auf die MS–DOS–Befehlsebene zurück.

Copying *t* tracks
***n* Sectors/Track, *s* Sides** (Kopiert werden *t* Spuren, *n* Sektoren/Spur, *s* Seiten)
[DISKCOPY]
DISKCOPY zeigt diese Meldung während des Disk–Duplizierungsvorganges an.

Copy process ended (Kopiervorgang abgebrochen)
[DISKCOPY]
Wegen eines fatalen Fehlers wurde der Duplizierungsvorgang abgebrochen. Verwenden Sie COPY oder XCOPY, um spezielle Dateien zu kopieren.

Copyright 1981,82,83,84,85,86 Microsoft Corp.
[MS–DOS]
Bestandteil der Startmeldung einiger Hilfsprogramme der MS–DOS–Systemdiskette.

Corrections will not be written to disk (Korrekturen werden nicht auf Disk geschrieben)
[CHKDSK]
Diese Meldung zeigt CHKDSK an, wenn Fehler auf der Disk gefunden werden, Sie aber beim Aufruf nicht den /F–Schalter gesetzt hatten. Wenn Fehler korrigiert werden sollen, müssen Sie CHKDSK mit diesem Schalter aufrufen.

Current date is *tt–mm–yy* (Aktuelles Datum ist *tt–mm–yy*)
[DATE]
Der DATE–Befehl zeigt zunächst das aktuelle Datum an. Das Datumsformat wird vom COUNTRY–Befehl in der CONFIG.SYS–Datei (s. Kapitel 2.3) bestimmt. Das Datum kann dann neu eingegeben werden oder durch Drücken der Eingabetaste bestätigt werden.

Current time is *hh.mm.ss,nn* (Aktuelle Zeit ist *hh.mm.ss,nn*)
[TIME]
Der TIME–Befehl zeigt zunächst die aktuelle Zeit in Stunden, Minuten, Sekunden und hundertstel Sekunden an. Das Zeitformat wird vom COUNTRY–Befehl in der CONFIG.SYS–Datei (s. Kapitel 2.3) bestimmt. Die Zeit kann dann neu eingegeben werden oder durch Drücken der Eingabetaste bestätigt werden.

Data error reading drive *x:* (Datenfehler beim Lesen von Laufwerk *x:*)
[MS–DOS]
Dies ist eine Geräte–Fehlermeldung (s. Kapitel A.1).

Delete current volume label (Y/N)? (Aktuelle Diskkennung löschen (J/N)?)
[LABEL]
Wenn auf der angegebenen Disk eine Diskkennung eingetragen ist, zeigt sie der LABEL–Befehl an. Wird dann nur die Eingabtaste gedrückt, zeigt LABEL diese Meldung. Wird mit Y (Yes=Ja) geantwortet, so wird die Kennung gelöscht, bei N (No=Nein) bleibt sie unverändert.

DEVICE Support Not Present (DEVICE–Unterstützung nicht vorhanden)
[DISKCOMP][DISKCOPY]
Das Disk–Laufwerk unterstützt nicht die Gerätesteuerung von MS–DOS Version 4.0.

Directory is joined (Verzeichnis ist verknüpft)
[CHKDSK]
CHKDSK überprüft keine Verzeichnisse, die mit dem JOIN–Befehl mit einem Laufwerk verknüpft wurden. Rufen Sie zum Auflösen der Verknüpfung JOIN mit dem /D–Schalter auf und starten Sie danach CHKDSK erneut.

Directory ist totally empty
no . or . . (Verzeichnis ist vollständig leer, keine .– oder . .–Einträge)
[CHKDSK]
Das angegebene Verzeichnis enthält keine Referenzen auf das aktuelle und das Vaterverzeichnis. Löschen Sie mit RD das Verzeichnis und erzeugen Sie es mit MD neu.

Directory not empty (Verzeichnis nicht leer)
[JOIN]
Sie können eine Verknüpfung zwischen einem Laufwerk und einem Verzeichnis mit JOIN nur herstellen, wenn das Verzeichnis leer ist.

Disk error reading/writing drive *x:* (Diskfehler beim Lesen/Schreiben auf Laufwerk *x:*)
[MS–DOS]
Dies ist eine Geräte–Fehlermeldung (s. Kapitel A.1).

Disk error reading/writing FAT (Diskfehler beim Lesen/Schreiben der Dateizuordnungstabelle)
[MS–DOS]
Eine der Dateizuordnungstabellen enthält einen defekten Sektor. MS–DOS benutzt dann automatisch die andere Tabelle. Sie sollten zunächst alle

Dateien auf eine andere Disk kopieren und anschließend den CHKDSK–Befehl mit dem Schalter /F aufrufen. Damit wird der Fehler dann korrigiert.

Diskette bad or incompatible (Diskette fehlerhaft oder inkompatibel)
[DISKCOPY]
Die Quelldisk ist nicht oder inkorrekt formatiert und kann deshalb nicht dupliziert werden.

Disk full. Edits lost (Disk voll. Edierter Text geht verloren)
[EDLIN]
EDLIN kann die edierte Datei wegen Speicherplatzmangel auf der Disk nicht mehr speichern. Ehe Sie den E–Befehl bei EDLIN geben, sollten Sie sicherstellen, daß Ihre Disk ausreichend freien Speicher für die edierte Datei enthält. Diese Disk sollte außerdem nicht schreibgeschützt sein.

Disk unsuitable for system disk (Disk als Systemdisk ungeeignet)
[FORMAT]
FORMAT hat beim Formatieren einer Disk an der Stelle, an der die Systemdateien gespeichert werden, einen fehlerhaften Sektor gefunden. Eine solche Disk kann nur für Datenspeicherung verwendet werden.

**Does *Name* specify a file name
or directory name on the target
(F=file D=directory)?** (Bezeichnet *Name* eine Datei oder ein Verzeichnis auf der Zieldisk (F = Datei D = Verzeichnis)?)
[XCOPY]
XCOPY zeigt diese Anfrage, wenn das Zielverzeichnis nicht existiert. Geben Sie F ein, wenn der angegebene Name eine Datei oder D, wenn er ein Verzeichnis bezeichnet, das noch nicht existiert.

(.)(..) Does not exist (.– oder . .–Eintrag existiert nicht)
[CHKDSK]
Diese informative Meldung zeigt CHKDSK an, wenn es bei der Überprüfung der Verzeichnisstruktur ungültige .– oder . .–Einträge in einem Verzeichnis findet.

**Do not specify filename(s)
Command format: DISKCOMP d: d:[/1][/8]** (Keine Dateinamen angeben. Befehlsformat: DISKCOMP d: d:[/1][/8])
[DISKCOMP]
Sie haben beim DISKCOMP–Befehl einen unkorrekten Schalter oder zu einer Laufwerksangabe zusätzlich eine Dateibezeichnung angegeben.

Do not specify filename(s)
Command format: DISKCOPY d: d:[/1][/8] (Keine Dateinamen angeben. Befehlsformat: DISKCOPY d: d:[/1][/8])
[DISKCOPY]
Sie haben beim DISKCOPY–Befehl einen unkorrekten Schalter oder zu einer Laufwerksangabe zusätzlich eine Dateibezeichnung angegeben.

DOS 2.0 or later required (MS–DOS Version 2.0 oder neuer erforderlich)
[ATTRIB][BACKUP][FC][GRAPHICS][JOIN][MODE][RESTORE][SUBST]
Diese Befehle oder Hilfsprogramme können nicht unter der MS–DOS-Version 1.xx benutzt werden.

Do you see the leftmost 0? (Y/N) (Sehen Sie die äußerst linke 0? (J/N))
[MODE]
Sie haben MODE mit den Parametern ,R,T aufgerufen und erhalten oben auf dem Bildschirm eine Zahlenreihe als Testbild angezeigt. Geben Sie Y (Yes = Ja) ein, wenn Sie die äußerst linke 0 sehen und N (No = Nein), wenn die Bildschirmanzeige nach rechts verschoben werden soll.

Do you see the rightmost 9? (Y/N) (Sehen Sie die äußerst rechte 9? (J/N))
[MODE]
Sie haben MODE mit den Parametern ,L,T aufgerufen und erhalten oben auf dem Bildschirm eine Zahlenreihe als Testbild angezeigt. Geben Sie Y (Yes = Ja) ein, wenn Sie die äußerst rechte 9 sehen und N (No = Nein), wenn die Bildschirmanzeige nach links verschoben werden soll.

Drive letter must be specified (Laufwerksbuchstabe muß angegeben werden)
[FORMAT]
Sie haben beim FORMAT–Befehl die Laufwerksangabe für die zu formatierende Disk vergessen.

Drive *x:* not ready
Make sure a diskette is inserted into the drive and the door is closed (Laufwerk *x:* nicht bereit. Stellen Sie sicher, daß eine Diskette eingelegt und das Laufwerk geschlossen ist)
[DISKCOMP][DISKCOPY]
Das eine Laufwerk enthält keine Diskette oder ist nicht geschlossen.

Drive types or diskette types not compatible (Laufwerks– oder Diskettentypen nicht kompatibel)
[DISKCOMP][DISKCOPY]
Beide Befehle verlangen Laufwerke und Disketten gleichen Typs. Z.B. kann nicht von einer einseitig formatierten auf eine doppelseitig formatierte Diskette kopiert werden oder umgekehrt oder es kann nicht eine High–Density–Diskette mit einer Double–Density–Diskette verglichen werden. Wenn Dateien auf zwei verschieden formatierten Disketten verglichen werden sollen, muß das Hilfsprogramm FC (s. Kapitel 9) verwendet werden. Kopien können mit den Befehlen COPY oder XCOPY erstellt werden.

Duplicate file name (Dateiname existiert bereits)
[RENAME]
Sie haben versucht, einer Datei einen neuen Namen zu geben, der in dem aktuellen Verzeichnis bereits für eine extistierende Datei vergeben wurde oder die angebene Datei existiert nicht.

ECHO ist off/on (ECHO ist AUS/AN)
[MS–DOS]
Diese informative Meldung zeigt an, ob der Schalter, der die Anzeige gerade bearbeiteter Stapelbefehle steuert, ein– oder ausgeschaltet ist, wenn ECHO ohne Parameter angegeben wird.

End of input file (Ende der Eingabedatei)
[EDLIN]
EDLIN hat die gesamte Datei in den Speicher gelesen. Wird die Datei in Teilen eingelesen, so zeigt EDLIN mit dieser Meldung an, daß der letzte Teil der Datei gelesen wurde.

Enter current Volume Label for drive *x:* (Aktuelle Diskkennung für Laufwerk *x:* eingeben)
[FORMAT]
FORMAT fordert mit dieser Anzeige zur Eingabe der Diskkennung auf, ehe mit der Formatierung der Festplatte begonnen wird. Wenn Sie die Diskkennung nicht wissen, brechen Sie FORMAT mit Ctrl–Break ab und lassen sich mit dem VOL–Befehl die Diskkennung für das betreffende Laufwerk anzeigen. Anschließend geben Sie den FORMAT–Befehl erneut.

Enter new date: (Neues Datum eingeben)
[DATE]
Sie haben sich mit dem DATE–Befehl das aktuelle Datum anzeigen lassen und werden nun aufgefordert, ein neues Datum einzugeben. Wenn Sie das aktuelle Datum bestätigen wollen, drücken Sie nur die Eingabetaste.

Enter new time: (Neue Zeit eingeben)
[TIME]
Sie haben sich mit dem TIME–Befehl die aktuelle Zeit anzeigen lassen und werden nun aufgefordert, eine neue Zeit einzugeben. Wenn Sie die aktuelle Zeit bestätigen wollen, drücken Sie nur die Eingabetaste.

Entry error (Eingabefehler)
[EDLIN]
Der zuletzt eingegebene EDLIN–Befehl war syntaktisch falsch. Geben Sie den Befehl in der richtigen Syntax ein und drücken Sie die Eingabetaste.

Entry has a bad attribute/link/size (Eintrag hat falsche(s) Attribut/Verknüpfung/Größe)
[CHKDSK]
Diese Meldung kann mit . oder .. beginnen, um zu zeigen, welches Unterverzeichnis fehlerhaft ist. Wenn Sie CHKDSK mit dem /F–Schalter aufgerufen haben, wird versucht, diesen Fehler automatisch zu korrigieren.

Error in .EXE file (Fehler in .EXE–Datei)
[MS–DOS]
Die unter MS–DOS aufgerufene .EXE–Datei hat ein ungültiges internes Format. Dieses Programm kann nicht gestartet werden. Prüfen Sie, ob Sie die richtige MS–DOS–Version benutzen.

Error reading/writing partition table (Fehler beim Lesen/Schreiben der Partitions–Tabelle)
[FORMAT]
FORMAT kann die Partitions–Tabelle nicht lesen oder schreiben. Sie sollten zunächst die Festplatte mit FDISK (s. Kapitel 3) neu partitionieren und dann erneut FORMAT aufrufen.

Errors found, F parameter not specified
Corrections will not be written to disk (Fehler aufgetreten, F–Schalter nicht gesetzt, Korrekturen werden nicht ausgeführt)
[CHKDSK]
Mit dieser informativen Meldung zeigt CHKDSK an, daß beim Überprüfen der angegebenen Disk Fehler aufgetreten sind, die nicht korrigiert werden, da /F nicht gesetzt wurde. Rufen Sie CHKDSK mit /F auf, um solche Fehler automatisch zu korrigieren.

Errors on list device indicate that it
may be off–line. Please check it. (Fehler beim List–Gerät zeigen, daß dieses nicht angeschlossen ist. Bitte überprüfen)

[PRINT]
Der Drucker ist nicht angeschlossen oder eingeschaltet oder steht auf OFF–LINE.

Error trying to open backup log file
Continuing without making log entries (Protokolldatei kann nicht geöffnet werden. BACKUP wird ohne Protokolleinträge fortgesetzt)
[BACKUP]
Sie haben BACKUP mit dem /L–Schalter aufgerufen, um eine Protokolldatei zu erstellen. Diese kann jedoch aus irgendwelchen Gründen (z.B. mangelnder Speicherplatz) nicht geöffnet werden.

Error writing to device (Schreibfehler bei einem Gerät)
[MS–DOS]
Sie haben versucht, zu viele Daten an ein Gerät zu senden, so daß MS–DOS diese Daten nicht zum Schreiben an das Gerät weitergeben kann. Reduzieren Sie die Datenmenge und wiederholen Sie die Ausgabe.

EXEC failure (EXEC–Fehler)
[MS–DOS]
MS–DOS hat entweder beim Lesen eines Befehls einen Fehler gefunden oder die Anzahl der mit dem FILES–Befehl in der CONFIG.SYS–Datei gesetzten Dateipuffer ist zu klein. Geben Sie mit FILES einen größeren Wert an (s. Kapitel 2.3) und starten Sie MS–DOS neu.

FCB unavailable reading/writing drive *x:* (Dateisteuerblock (FCB) beim Lesen/Schreiben auf Laufwerk *x:* nicht verfügbar)
[MS–DOS]
Dies ist eine Geräte–Fehlermeldung (s. Kapitel A.1).

fc:cannot open *Dateiname* – No such file or directory
(FC: *Dateiname* kann nicht geöffnet werden – keine solche Datei oder kein Verzeichnis)
[FC]
Eine der beim FC–Programm angegebenen Dateien existiert nicht. Benutzen Sie den DIR–Befehl, um die korrekte Schreibweise der Datei zu überprüfen.

fc: *Dateiname* longer than *Dateiname* (FC: *Dateiname* länger als *Dateiname*)
[FC]
FC hat beim Vergleich zweier Dateien, das Ende der einen Datei erreicht, ohne daß die zweite Datei zuende ist.

fc: incompatible switches (FC:unkompatible Schalter gesetzt)
[FC]
Sie haben beim Aufruf von FC unkompatible Schalter (z.B. /B und /L) gesetzt. Binär– und ASCII–Vergleich sind nicht kombinierbar.

fc:no differences encountered (FC:keine Unterschiede gefunden)
[FC]
FC zeigt mit dieser informativen Meldung an, daß die beiden verglichenen Dateien identisch sind.

fc:out of memory (FC:Speicher nicht ausreichend)
[FC]
Der Hauptspeicher reicht für den Dateivergleich nicht aus.

File allocation table bad (Dateizuordnungstabelle fehlerhaft)
[MS–DOS]
Die gerade benutzte Disk kann fehlerhaft sein. Starten Sie CHKDSK /F, um die Disk zu überprüfen und ggf. zu reparieren.

File allocation table bad drive *x:* (Dateizuordnungstabelle auf Laufwerk *x:* fehlerhaft)
[CHKDSK]
Die überprüfte Disk im angegebenen Laufwerk ist gar nicht oder nicht korrekt formatiert, oder sie enthält ein anderes Betriebssystem.

File *Dateiname* **canceled by operator** (Datei *Dateiname* vom Bediener gelöscht)
[PRINT]
Sie haben PRINT mit dem /T–Schalter aufgerufen, um die Druck–Warteschlange zu leeren. PRINT bestätigt diesen Vorgang mit dieser Meldung.

File cannot be converted (Datei kann nicht umgewandelt werden)
[EXE2BIN]
Die beim EXE2BIN–Befehl angegebene Datei hat nicht das korrekte Format.

File cannot be copied onto itself (Datei kann nicht auf sich selbst kopiert werden)
[COPY][REPLACE][XCOPY]
Sie haben für Quell– und Zieldatei denselben Namen angegeben.

File creation error (Dateierzeugungsfehler)
[MS–DOS][EDLIN][RESTORE][XCOPY]
Sie haben versucht, einen neuen Dateinamen anzugeben oder eine Datei zu ersetzen, die bereits in diesem Verzeichnis existiert, oder der Speicherplatz reicht für die Datei nicht aus. Wenn die Datei bereits existiert, ist es eine schreibgeschützte (read–only) Datei, die nicht ersetzt werden kann. Dieser Fehler kann auch angezeigt werden, wenn das Basisverzeichnis keine Dateieinträge mehr erlaubt oder wenn der Dateiname derselbe wie eine Diskkennung, ein Verzeichnisname oder der Name einer verborgenen Datei ist.

File is READ–ONLY (Datei ist schreibgeschützt)
[EDLIN]
Sie haben EDLIN mit einer schreibgeschützten Datei aufgerufen, die nicht geändert werden kann. Überlegen Sie genau, ob Sie diese Datei ändern wollen und löschen Sie dann ggf. das Schreibschutzattribut mit dem ATTRIB–Befehl.

File name must be specified (Dateiname muß angegeben werden)
[EDLIN]
EDLIN kann nur mit einem Dateinamen aufgerufen werden. Geben Sie nach dem Befehlswort EDLIN den Namen der zu edierenden Datei ein und drücken Sie dann die Eingabetaste.

File not found (Datei nicht gefunden)
[CHKDSK][EDLIN][FC][FIND][PRINT][RECOVER][RENAME][XCOPY]
MS–DOS kann die angegebene Datei nicht finden oder Sie haben versucht, einer Datei mit RENAME einen Namen zu geben, der in dem aktuellen Verzeichnis bereits für eine andere Datei oder ein anderes Verzeichnis vergeben ist.

File not in PRINT queue (Datei nicht in PRINT–Warteschlange)
[PRINT]
Sie haben versucht, eine Datei aus der PRINT–Warteschlange zu entfernen, die dort nicht verzeichnet ist. Überprüfen Sie, ob Sie den Dateinamen richtig eingegeben haben.

Files cannot be added to this diskette
Unless the PACK (/P) switch is used
Set the switch (Y/N)? (Dateien können auf dieser Diskette nicht hinzugefügt werden, wenn der /P–Schalter nicht gesetzt wird. Schalter setzen (J/N)?)
[BACKUP]

Die Zieldiskette hat nicht ausreichend Speicherplatz zur Verfügung, um die Dateien der Quelldiskette ungeteilt aufzunehmen. Wenn Sie die Aufteilung großer Dateien über die Grenze einer Diskette hinweg nicht wollen, müssen Sie mit N (No = Nein) antworten. Bei Y (Yes = Ja) werden die Disketten so voll wie möglich geschrieben, wobei große Dateien ggf. auf der Folgediskette fortgesetzt werden.

***** Files were backed up at *Zeit* on *Datum* ***** (*** Dateien wurden am *Datum* um *Zeit* gesichert ***)
[RESTORE]
Diese informative Meldung zeigt RESTORE beim Rückschreiben von mit BACKUP gesicherten Dateien an.

FIND: Access denied (FIND: Zugriff verweigert)
[FIND]
Der Zugriff auf die angegebene Datei ist nicht möglich. Stellen Sie sicher, daß die Disk oder die Datei nicht schreibgeschützt oder bei Netzwerkbetrieb gegen Zugriff verriegelt ist.

FIND: File not found (FIND: Datei nicht gefunden)
[FIND]
MS–DOS kann die beim FIND–Befehl angegebene Datei nicht finden. Überprüfen Sie die Schreibweise des Dateinamens.

FIND: Invalid number of Parameters (FIND: Ungültige Anzahl von Parametern)
[FIND]
Sie haben zuviele oder zuwenige Parameter beim Aufruf des FIND–Befehls angegeben.

FIND: Invalid Parameter (FIND: Ungültiger Parameter)
[FIND]
Einer der beim Aufruf von FIND angegebnen Schalter ist falsch.

FIND: Read error in *Dateiname* (FIND: Lesefehler bei *Dateiname*)
[FIND]
FIND kann die angegebene Datei nicht lesen.

FIND: Syntax error (FIND: Syntaxfehler)
[FIND]
Überprüfen Sie den FIND–Aufruf auf korrekte Syntax.

First cluster number is invalid, entry truncated (Erste Bereichsnummer ungültig, Eintrag gekürzt)
[CHKDSK]
Der Dateiverzeichnis–Eintrag enthält einen ungültigen Zeiger auf den Datenbereich. Wenn Sie CHKDSK mit dem /F–Schalter aufgerufen haben, wird die Datei auf die Länge Null gekürzt.

FIRST diskette bad or incompatible (Erste Diskette fehlerhaft oder inkompatibel)
[DISKCOMP]
DISKCOMP kann das Format der Quelldiskette nicht identifizieren. Starten Sie CHKDISK, um den Fehler zu diagnostizieren.

Fixups needed – base segment *hex:* (Segmentfixierungen erforderlich – Basissegment *hex:*)
[EXE2BIN]
Die Quell (.EXE)–Datei enthält Informationen, die ein Ladesegment für diese Datei erforderlich machen. Sie müssen eine absolute Segmentadresse angeben, bei der der umgewandelte Modul plaziert werden soll.

For cannot be nested (FOR–Befehl kann nicht geschachtelt werden)
[MS–DOS]
Die Schachtelung von FOR–Befehlen in einer Stapeldatei ist nicht erlaubt.

Format another (Y/N)? (Noch eine Diskette formatieren (J/N)?)
[FORMAT]
Diese Anfrage erscheint am Ende einer Diskettenformatierung. Geben Sie Y (Yes = Ja) ein, wenn eine weitere Diskette formatiert werden soll oder N (No = Nein), wenn beendet werden soll. Haben Sie irrtümlich Y gegeben, so können Sie bei der anschließenden Aufforderung **strike any key** mit Ctrl–Break abbrechen.

Format complete (Formatierung abgeschlossen)
[FORMAT]
Mit dieser informativen Meldung zeigt MS–DOS den Abschluß der Formatierung an.

Format failure (Formatierungsfehler)
[FORMAT]
MS–DOS kann die Disk nicht formatieren. Dieser Meldung wird im allgemeinen noch der Grund hinzugefügt.

Format not supported on drive *x:* (Diskformat auf Laufwerk *x:* nicht unterstützt)
[FORMAT]
FORMAT kann die Disk im angegebenen Laufwerk nicht formatieren. Für das angegebene Laufwerk wurden Geräteparameter spezifiziert, die der aktuelle FORMAT–Befehl nicht unterstützt (s. DRIVER.SYS–Beschreibung in Kapitel 2.5.2).

Formatting while copying (Formatieren während des Kopierens)
[DISKCOPY]
Mit dieser informativen Meldung zeigt DISKCOPY, daß die Disk–Duplizierung Formatieren und Kopieren in einem Arbeitsgang erledigt, wenn die Zieldiskette vorher noch nicht formatiert war.

General failure reading/writing drive *x:* (Allgemeiner Fehler beim Lesen/Schreiben auf Laufwerk *x:*)
[MS–DOS]
Dies ist eine Gerätefehlermeldung (s. Kapitel A.1).

Graftabl needs DOS version 2.0 or later (GRAFTABL benötigt MS–DOS Version 2.0 oder jünger)
[GRAFTABL]
GRAFTABL kann nicht mit MS–DOS–Versionen 1.xx benutzt werden.

Graphics characters already loaded (Grafischer Zeichensatz bereits geladen)
[GRAFTABL]
Diese informative Meldung wird angezeigt, wenn Sie den GRAFTABL-Befehl ein zweites Mal benutzen, da der Zeichensatz resident im Speicher bleibt.

Graphics characters loaded (Grafischer Zeichensatz geladen)
[GRAFTABL]
Diese informative Meldung wird angezeigt, wenn Sie den GRAFTABL-Befehl das erste Mal benutzen. Der Zeichenzatz bleibt dann resident im Speicher.

Has invalid cluster, file truncated (Hat ungültigen Bereich, Datei gekürzt)
[CHKDSK]
Der Dateiverzeichnis–Eintrag enthält einen ungültigen Zeiger auf den Datenbereich. Wenn Sie CHKDSK mit dem /F–Schalter aufgerufen haben, wird die Datei auf die Länge Null gekürzt.

Head: *hhh* Cylinder: *ccc* (Kopf *hhh*, Zylinder *ccc*)
[FORMAT]
Während der Formatierung gibt FORMAT laufend Kopf– und Zylindernummer aus, um den Fortgang der Formatierung anzuzeigen.

Illegal device name (Ungültiger Gerätename)
[MODE]
Sie haben ein Gerät angegeben, das MS–DOS nicht kennt.

Incompatible system size (Inkompatible Systemgröße)
[SYS]
Die Systemdateien auf der Quelldiskette belegen mehr Speicherplatz, als auf der Zieldiskette verfügbar ist. Sie können mit SYS auf die angegebene Zieldisk keine Systemdateien übertragen. Formatieren Sie eine neue Diskette mit FORMAT /S.

Incorrect DOS Version (Falsche MS–DOS–Version)
[APPEND][ATTRIB][BACKUP][CHKDSK][DISKCOMP][DISKCOPY]
[EDLIN][FC][FIND][FORMAT][GRAPHICS][JOIN][KEYB*xx*]
[LABEL][MODE][MORE][PRINT][RECOVER][REPLACE][RESTORE]
[SHARE][SORT][SUBST][SYS][TREE][XCOPY]
Einige externe MS–DOS–Befehle und –Hilfsprogramme können unter älteren Versionen von MS–DOS nicht benutzt werden und einige laufen nur exakt mit der Version, für die sie programmiert wurden. Sie müssen die korrekte MS–DOS–Version für den benutzten Befehl verwenden.

incorrekt number of parameters (Falsche Parameteranzahl)
[JOIN][SUBST]
Sie haben entweder zu viele oder zu wenige Parameter in der Befehlszeile angegeben.

Incorrect parameter (Falscher Parameter)
[ASSIGN][SHARE]
Einer der in der Befehlszeile angegebenen Parameter ist falsch.

Infinite retry on parallel printer timeout (Zeitüberschreitung beim Paralleldrucker)
[MODE]
Ihr Drucker steht möglicherweise auf OFF–LINE oder ist nicht bereit. Falls doch, müssen Sie den Computer durch gleichzeitiges Drücken der Tasten **Alt Ctrl Del** neu starten.

Insert backup diskett *n* into drive *x:* (BACKUP–Diskette *n* in Laufwerk *x:* einlegen)
[BACKUP][RESTORE]
Beim Sichern oder Zurückspeichern werden Sie aufgefordert, die angegebene Sicherungsdiskette einzulegen. Achten Sie darauf, jede BACKUP-Diskette in der richtigen Reihenfolge zu beschriften, damit das Rückspeichern erleichtert wird.

Insert destination disk in drive *x:*
and strike any key when ready (Zieldiskette einlegen. Wenn bereit, beliebige Taste drücken)
[SYS]
Diese Meldung erscheint, wenn Sie das Betriebssystem auf einem Computer mit nur einem Laufwerk übertragen wollen. Legen Sie die richtige Diskette ein und drücken Sie eine beliebige Taste.

Insert disk in drive *x:*
and strike any key when ready (Diskette einlegen. Wenn bereit, beliebige Taste drücken)
[MS–DOS]
Diese Meldung erscheint, wenn MS–DOS kopiert oder formatiert. Legen Sie die richtige Diskette ein und drücken Sie eine beliebige Taste.

Insert diskette with batch file
and press any key when ready (Diskette mit Stapeldatei einlegen. Wenn bereit, beliebige Taste drücken)
[MS–DOS]
Diese Meldung erscheint, wenn die Stapeldatei nicht auf der Disk im aktuellen Laufwerk enthalten ist. Legen Sie die richtige Diskette ein und drücken Sie eine beliebige Taste.

Insert DOS–Diskette in drive *x:*
and strike ENTER when ready (MS–DOS–Systemdiskette in Laufwerk *x:* einlegen. Wenn bereit, Eingabetaste drücken)
[FORMAT]
Sie haben den FORMAT /S –Befehl eingegeben. Die Diskette im aktuellen Laufwerk enthält jedoch nicht die Systemdateien. Legen Sie die richtige Diskette ein und drücken Sie die Eingabe–Taste.

Insert FIRST Diskette into drive *x:* (Erste Diskette in Laufwerk *x:* einlegen)
[DISKCOMP]
Mit dieser Meldung werden Sie zum Einlegen der ersten Diskette, die sie vergleichen wollen, aufgefordert.

Insert last backup diskette in drive x:
Strike any key when ready (Letzte BACKUP–Diskette in Laufwerk *x:* einlegen. Wenn bereit, beliebige Taste drücken)
[BACKUP]
BACKUP fordert Sie zum Einlegen der letzten Sicherungsdiskette auf. Legen Sie die richtige Diskette ein und drücken Sie eine beliebige Taste.

Insert restore target diskette in drive x: (Legen Sie die Zieldiskette für RESTORE in Laufwerk *x:* ein)
[RESTORE]
Wenn Sie Sicherungsdateien auf eine Diskette zurückspeichern wollen, fordert Sie RESTORE mit dieser Meldung zum Einlegen der Zieldiskette auf.

Insert SECOND Diskette into drive x: (Zweite Diskette in Laufwerk *x:* einlegen)
[DISKCOMP]
Mit dieser Meldung werden Sie zum Einlegen der zweiten Diskette, die sie mit der ersten vergleichen wollen, aufgefordert.

Insert source disk (Quelldiskette einlegen)
[BACKUP]
BACKUP fordert auf, die Quelldiskette mit den zu sichernden Dateien einzulegen.

Insert SOURCE Diskette into drive x: (Quelldiskette in Laufwerk *x:* einlegen)
[DISKCOPY]
Mit dieser Meldung werden Sie zum Einlegen der zu duplizierenden Diskette aufgefordert.

Insert system diskette in drive x:
and strike any key when ready (Systemdiskette in Laufwerk *x:* einlegen. Wenn bereit, beliebige Taste drücken)
[SYS]
SYS benötigt eine Diskette, von der es die beiden verborgenen Systemdateien MSDOS.SYS und IO.SYS lesen kann. Legen Sie eine Systemdiskette ein und drücken Sie eine beliebige Taste.

Insert TARGET diskette into drive x: (Zieldiskette in Laufwerk *x:* einlegen)
[DISKCOPY]
DISKCOPY fordert Sie auf, die Zieldiskette in das angegebene Laufwerk einzulegen. Verfügen Sie über einen Computer mit nur einem Diskettenlaufwerk, erhalten Sie diese Meldung ebenfalls.

Insufficient disk space (Ungenügender Disk–Speicherplatz)
[MS–DOS][REPLACE][SORT][XCOPY]
Die bezogene Disk ist nahezu voll und enthält deshalb nicht mehr ausreichend Speicherplatz, um die gerade bearbeitete Funktion auszuführen.

Insufficient memory (Ungenügender Speicherplatz)
[BACKUP][CHKDSK][DISKCOMP][DISKCOPY][EDLIN][REPLACE]
[RESTORE][SORT][XCOPY]
Ihr Computer verfügt nicht mehr über ausreichend freien Hauptspeicher, um die gerade bearbeitete Funktion auszuführen. Ehe Sie den Befehl wiederholen, müssen Sie, falls möglich, freien Hauptspeicherplatz durch Löschen von speicherresidenten Dateien schaffen. Bei EDLIN geht das mit dem W–Befehl, gefolgt von einem A–Befehl.

Insufficient memory for systemtransfer (Ungenügender Speicherplatz für Systemübertragung)
[FORMAT]
Ihre Speicherkonfiguration reicht für die Übertragung des Systems mit dem FORMAT /S–Befehl nicht aus.

Insufficient room in root directory.
Erase files in root and repeat CHKDSK (Ungenügend Platz im Basisverzeichnis. Dateien löschen und CHKDSK erneut aufrufen)
[CHKDSK]
CHKDSK rettet grundsätzlich verlorengegangene Dateien in das Basisverzeichnis. In diesem Fall kann Ihr Basisverzeichnis keine weiteren Einträge mehr aufnehmen. Löschen Sie dort nicht mehr benötigte Dateien oder verlagern Sie diese in ein anderes Verzeichnis, um Platz zu schaffen.

Intermediate file error during pipe (Zwischendateifehler bei Filterbefehl)
[MS–DOS]
Der benutzte Filterbefehl benötigt temporäre Dateien auf der Disk, die nach Beendigung des Filterbefehls automatisch gelöscht werden. In einer solchen Datei ist ein Fehler aufgetreten. Stellen Sie sicher, daß die Disk über ausreichend Speicherkapazität für solche temporären Dateien verfügt und daß die Disk nicht schreibgeschützt ist. Wiederholen Sie dann den Befehl.

Internal error (Interner Fehler)
[FC][MODE][SHARE]
Der benutzte externe Befehl oder das Hilfsprogramm enthält einen Fehler. In diesem Fall müssen Sie Rücksprache mit Ihrem Vertragshändler nehmen.

Invalid argument (Ungültiges Argument)
[BACKUP][FC][RESTORE]
Beim angegeben Befehl haben Sie ein ungültiges Argument (Parameter) benutzt. Überprüfen Sie mit Hilfe der Befehlsbeschreibung im Kapitel 6.3.1 die korrekte Befehlssyntax.

Invalid baud rate specified (Ungültige Übertragungsgeschwindigkeit)
[MODE]
Bei der Konfigurierung der Datenfernübertragungsschnittstelle (serielle Schnittstelle) haben Sie eine ungültige Baud–Rate angegeben. Erlaubte Baudraten sind: 110, 150, 300, 600, 1200, 2400, 4800 und 9600. Sie müssen zumindest die ersten beiden Ziffern der Baud–Rate angeben.

Invalid characters in volume label (Ungültige Zeichen in der Diskkennung)
[FORMAT][LABEL]
Bei der Angabe der Diskkennung haben Sie ein oder mehrere ungültige Zeichen verwendet. Sie dürfen nur 11 Zeichen und nur die auch für Dateinamen erlaubten Zeichen verwenden (s.a. Kapitel 4.1.5).

Invalid COMMAND.COM
Insert COMMAND.COM disk in default drive
and strike any key when ready (Ungültige COMMAND.COM–Datei. COMMAND.COM–Disk in aktuelles Laufwerk einlegen. Wenn bereit, beliebige Taste drücken)
[MS–DOS]
Das gerade ausgeführte Programm hat nahezu den gesamten verfügbaren Hauptspeicher benötigt, so daß MS–DOS jetzt den Befehlsprozessor COMMAND.COM neu laden muß. Entweder findet MS–DOS diese Datei nicht oder die gefundene Kopie ist nicht die korrekte Version. Legen Sie eine Systemdiskette mit COMMAND.COM ein und drücken Sie eine beliebige Taste.

Invalid country code (Ungültiger Ländercode)
[MS–DOS]
Sie haben in Ihrer CONFIG.SYS–Datei beim COUNTRY–Befehl einen ungültigen Ländercode, der Datum– und Zeitformat sowie Dezimalzeichen und Währungssymbol festlegt, benutzt. In Kapitel 2.3 sind die erlaubten COUNTRY–Codes aufgeführt. Für Deutschland gilt der Code 049.

Invalid current directory (Ungültiges aktuelles Verzeichnis)
[CHKDSK]
Die Diskette im aktuellen Laufwerk enthält ein ungültiges Verzeichnis.

Möglicherweise können Sie einige Dateien dieser Disk mit dem COPY-Befehl retten. Andernfalls ersetzen Sie sie durch eine neue Diskette.

Invalid date (Ungültiges Datum)
[DATE][XCOPY]
In Beantwortung der Datumsanforderung haben Sie ein ungültiges Datum oder Datumsformat angegeben. Im Kapitel 6.3.1 beim DATE–Befehl ist die korrekte Datumssyntax angegeben.

Invalid Date/Time (Ungültiges Datum/Zeit)
[BACKUP]
Bei einem der BACKUP–Schalter haben Sie ein ungültiges Format für Datum und/oder Zeit angegeben. Im Kapitel 6.3.1 beim BACKUP–Befehl ist die richtige Syntax beschrieben.

Invalid device (Ungültiges Gerät)
[MS–DOS]
Das angegebene Gerät war nicht AUX, CON, NUL oder PRN.

Invalid device parameters from device driver (Ungültige Geräteparameter vom Gerätetreiber)
[FORMAT]
Die Anzahl der verborgenen Sektoren ist nicht ohne Rest teilbar durch die Anzahl von Sektoren/Spur; d.h., die Partition beginnt nicht an einer Spurgrenze. Dies kann passieren, wenn Sie eine Festplatte, die vorher unter MS–DOS, Version 2.*xx*, formatiert wurde, neu formatieren, ohne sie vorher mit FDISK neu zu partitionieren, oder wenn Sie die Gerätetreiber–Parameter falsch gesetzt haben. Überprüfen Sie Ihre CONFIG.SYS–Datei auf falsche DEVICE– oder DRIVPARM–Parameter (s. Kapitel 2.3).

Invalid directory (Ungültiges Verzeichnis)
[MS–DOS]
Das angegebene Verzeichnis ist entweder ungültig oder existiert nicht. Überprüfen Sie den eingegebenen Verzeichnisnamen auf korrekte Schreibweise.

Invalid disk change reading/writing drive *x:* (Unerlaubter Diskettenwechsel beim Lesen/Schreiben auf Laufwerk *x:*)
[MS–DOS]
Dies ist eine Gerätefehlermeldung (s. Kapitel A.1).

Invalid drive in search path (Ungültiges Laufwerk im Suchpfad)
[MS–DOS]
Das im Suchpfad angegebene Laufwerk existiert nicht.

Invalid drive or filename (Ungültiges Laufwerk oder ungültiger Dateiname)
[EDLIN][RECOVER]
Laufwerks– oder Dateiangabe sind ungültig. Überprüfen Sie die Eingabe auf richtige Schreibweise.

Invalid drive specification (ungültige Laufwerksangabe)
[BACKUP][CHKDSK][DISKCOMP][DISKCOPY][FORMAT][LABEL]
[PRINT][REPLACE][RESTORE][SYS][TREE][XCOPY]
Die Laufwerksangabe ist ungültig oder bezieht sich auf ein nicht existierendes Laufwerk. Überprüfen Sie Ihre Eingabe.

Invalid environment size specified (Ungültige Größe für Umfeld–Zeichentabelle angegeben)
[COMMAND]
Beim COMMAND–Befehl haben Sie eine ungültige Byte–Anzahl beim /E–Schalter angegeben. Erlaubt sind Werte zwischen 128 und 32768.

Invalid number of parameters (Ungültige Parameter–Anzahl)
[ATTRIB][BACKUP][FC][FIND][RECOVER][RESTORE][XCOPY]
Entweder haben Sie eine erforderliche Befehlsoption oder Zeichenkette nicht angegeben, oder Sie haben in der Befehlszeile einen ungültigen Parameter angegeben.

Invalid parameter(s) (Ungültiger oder ungültige Parameter)
[BACKUP][CHKDSK][DISKCOMP][DISKCOPY][EDLIN][FIND]
[FORMAT][JOIN][MODE][PRINT][REPLACE][RESTORE][SORT]
[SUBST][SYS][TREE][XCOPY]
Einer der in der Befehlszeile angegebenen Schalter ist falsch oder existiert nicht. Im Kapitel 6.3.1 beim jeweiligen Befehl ist die korrekte Syntax beschrieben.

Invalid path,not directory, or directory not empty (Ungültiger Pfad, kein Verzeichnis oder Verzeichnis nicht leer)
[MS–DOS]
Das angegebene Verzeichnis kann aus den angezeigten Gründen nicht gelöscht werden.

Invalid path (or file not found) (Ungültiger Pfad (oder Datei nicht gefunden))
[ATTRIB][BACKUP][COPY][RESTORE][TREE][XCOPY]
Sie haben eine Pfad oder Dateinamen angegeben, der nicht existiert. Überprüfen Sie Ihre Eingabe.

Invalid sub-directory entry (Ungültiger Unterverzeichnis–Eintrag)
[CHKDSK]
Das angegebene Unterverzeichnis existiert nicht oder ist ungültig. Achten Sie auf korrekte Schreibweise.

Invalid time (ungültige Zeit)
[TIME]
Beim TIME–Befehl haben Sie nach der Anzeige der aktuellen Zeit die neue Zeit in einem falschen Format eingegeben. In Kapitel 6.3.1 beim TIME–Befehl ist das richtige Format beschrieben.

Invalid Volume ID (ungültige Diskkennung)
[FORMAT]
Sie haben bei der Neuformatierung der Festplatte eine Diskkennung eingegeben, die nicht mit der bereits vorhandenen übereinstimmt. Der Formatbefehl wird bei dieser Meldung beendet. Benutzen Sie den VOL–Befehl, um die aktuelle Kennung anzuzeigen und versuchen Sie es dann noch einmal.

Label not found (Sprungmarke nicht gefunden)
[MS–DOS]
Die gerade abgearbeitete Stapeldatei enthält einen GOTO–Verzweigungsbefehl mit einer Marke, die in der Stapeldatei nicht existiert. Korrigieren Sie die Stapeldatei (z.B. mit EDLIN).

Last backup diskette not inserted
Insert last backup diskette in drive *x:*
Strike any key when ready (Letzte Backup–Diskette nicht eingelegt. Bitte in Laufwerk *x:* einlegen. Wenn bereit, beliebige Taste drücken)
[BACKUP]
Sie werden aufgefordert, bei einem Sicherungslauf die letzte Sicherungsdiskette in das angegebene Laufwerk einzulegen und dann eine beliebige Taste zu drücken.

***** Last file not backed up ***** (*** Letzte Datei nicht gesichert ***)
[BACKUP]

BACKUP kann die letzte Datei der Disk nicht sichern. Der Grund kann nicht ausreichender Speicher auf der Zieldisk oder ein Fehler in der Quelldatei oder auf der Zieldisk sein.

Line too long (Zeile zu lang)
[EDLIN]
Während der Ausführung eines R–Befehls (Ersetze) wird wegen der Länge der zu ersetzenden Zeichenkette die Zeile zu lang (mehr als 253 Zeichen). Teilen Sie sie in zwei Zeilen und versuchen Sie den R–Befehl erneut.

List output is not assigned to a device (Druckausgabe geht nicht an ein Gerät)
[PRINT]
Beim Aufruf des PRINT–Befehls fordert MS–DOS die Angabe eines Druckers. Sie erhalten diese Meldung, wenn Sie den PRINT–Befehl für ein Gerät aufsetzen, das nicht existiert.

Lock violation reading/writing drive *x:* (Dateizugriff beim Lesen/Schreiben auf Laufwerk *x:* nicht möglich)
[MS–DOS]
Dies ist eine Gerätefehlermeldung (s. Kapitel A.1).

***x* lost clusters found in *y* chains**
Convert lost chaines to files (Y/N)? (*x* verlorene Bereiche in *y* Blöcken gefunden. Verlorene Blöcke in Dateien umwandeln (J/N)?)
[CHKDSK]
CHKDSK hat bei der Disküberprüfung Blöcke gefunden, die zwar zugewiesen sind, aber nicht mit einer der Dateien verknüpft werden konnten. Wird mit Y (Yes = Ja) geantwortet, so wird für jede Zeichenkette ein Dateieintrag der Form FILE*nnnn*.CHK erstellt, beginnend mit *nnnn* = 0001. Wenn Sie CHKDSK nicht mit dem /F–Schalter aufgerufen haben, erhalten Sie die Anzeige ***x* Bytes would be freed** (*x* Bytes würden freigemacht). Wird mit N (No = Nein) geantwortet, so werden diese Blöcke freigegeben und die darin enthaltenen Daten sind verloren. Wenn Sie CHKDSK nicht mit dem /F–Schalter aufgerufen haben, passiert gar nichts.

LPT*n*: not redirected (LPT*n*: nicht umgeleitet)
[MODE]
MODE kann die Ausgabe über die Paralleldruckerschnittstelle nicht umleiten. Überprüfen Sie, ob Sie die richtigen Parameter angegeben haben.

LPT*n*: redirected to COM*m*: (LPT*n*: auf COM*m*: umgeleitet)
[MODE]
Alle Druckerausgaben über die Parallel–Druckerschnittstelle LPT*n*: werden künftig über den Datenfernübertragungsanschluß COM*m*: (serielle Schnittstelle) umgeleitet.

LPT*n*: set for 80 (LPT*n*: auf 80 Zeichen/Zeile gesetzt)
[MODE]
Die Druckzeilenlänge des an LPT*n*: angeschlossenen Druckers ist auf 80 Zeichen gesetzt.

LPT*n*: set for 132 (LPT*n*: auf 132 Zeichen/Zeile gesetzt)
[MODE]
Die Druckzeilenlänge des an LPT*n*: angeschlossenen Druckers ist auf 132 Zeichen gesetzt.

Memory allocation error.
Cannot load MS–DOD, system halted (Speicherzuordnungs–Fehler. MS–DOS kann nicht geladen werden. System angehalten)
[MS–DOS]
Wenn Sie diese Meldung erhalten, machen Sie eine neue Kopie von Ihrer Systemdiskette und starten Sie das System neu.

––More–– (––Mehr––)
[MORE]
Der Filterbefehl MORE zeigt diese Meldung in der letzten Bildschirmzeile an und wartet mit der Ausgabe weiterer Daten, wenn der Bildschirm voll ist. Drücken einer beliebigen Taste setzt die Ausgabe fort.

MORE: Incorrect DOS version (MORE: Ungültige MS–DOS–Version)
[MORE]
Der MORE–Befehl läuft nicht mit MS–DOS–Versionen, die älter als Version 2.0 sind.

Must specify destination line number (Ziel–Zeilennummer muß angegeben werden)
[EDLIN]
Bei einem C (Kopiere)– oder einem M (Verschiebe)–Befehl haben Sie die Ziel–Zeilennummer vergessen. Geben Sie den Befehl mit einer Ziel–Zeilennummer erneut ein.

Must specify ON or OFF (ON oder OFF muß angegeben werden)
[MS–DOS]
Der Befehl (z.B. BREAK oder ECHO) benötigt ein Argument ON (EIN) oder OFF (AUS).

Name of list device [PRN]: (Name des Druckers [PRN]:)
[PRINT]
Wenn Sie PRINT das erste mal und ohne den /D–Schalter aufrufen, erhalten Sie diese Meldung. Sie können hier jeden Namen eines gültigen Ausgabegerätes angeben. Drücken Sie nur die Eingabetaste, so nimmt PRINT den Drucker PRN als Ausgabegerät.

New file (Neue Datei)
[EDLIN]
EDLIN zeigt diese Meldung, wenn die beim Aufruf angegebene Datei noch nicht existiert. Wollen Sie tatsächlich eine neue Datei erzeugen, können Sie diese Meldung ignorieren. Wenn nicht, überprüfen Sie den eingegebenen Namen auf richtige Schreibweise.

No appended directories (Keine angefügten Verzeichnisse)
[APPEND]
Sie haben beim APPEND–Befehl keinen Suchpfad angegeben.

No COM: ports (Keine COM:–Schnittstellen)
[MODE]
Ihr Computer verfügt über keinen Datenfernübertragungsanschluß (serielle Schnittstelle).

No files added/replaced (Keine Dateien hinzugefügt/ersetzt)
[REPLACE]
REPLACE hat keine Datei zum Ziel hinzugefügt oder dort ersetzt.

No files found *Dateiname* (Keine Dateien gefunden *Dateiname*)
[REPLACE]
REPLACE findet keine übereinstimmenden Quell– oder Ziel–Dateien mit dem angegebenen Namen.

No free handels.
Cannot start COMMAND.COM, exiting (Keine freien Dateinummern. COMMAND.COM kann nicht gestartet werden. Beendet)
[MS–DOS]
Wegen zu vieler offener Dateien kann eine Kopie von COMMAND.COM nicht geladen werden. Erhöhen Sie den Wert beim FILES–Befehl in der CONFIG-.SYS–Datei (s. Kapitel 2.3) und starten Sie MS–DOS neu.

Non-DOS disk error reading/writing drive *x:* (Keine MS–DOS–Disk beim Lesen/Schreiben auf Laufwerk *x:*)
[MS–DOS]
Dies ist eine Geräte–Fehlermeldung (s. Kapitel A.1).

Non-system disk or disk error
Replace and strike any key when ready (Keine System–Disk oder Disk–Fehler. Diskette wechseln. Wenn bereit, beliebige Taste drücken)
[FORMAT][SYS]
Ersetzen Sie die Diskette durch eine fehlerfreie Systemdiskette und drücken Sie eine beliebige Taste.

No paper error writing device *Gerät* (Kein Papier bei der Ausgabe auf Gerät *Gerät*)
[MS–DOS]
Dies ist eine Geräte–Fehlermeldung (s. Kapitel A.1).

No path (Kein Suchpfad gesetzt)
[PATH]
Sie erhalten diese informative Meldung, wenn Sie den PFAD–Befehl ohne Parameter eingeben und noch kein Suchpfad gesetzt ist.

No room for system on destination disk (Kein Platz für Systemdateien auf Zieldisk)
[SYS]
Die Zieldisk enthält nicht mehr ausreichend Speicherpatz für die Systemdateien MS-DOS.SYS und IO.SYS. Löschen Sie einige Dateien oder nehmen Sie eine andere Diskette. Möglicherweise müssen Sie eine neue Diskette formatieren (FORMAT mit /S–Schalter).

No room in directory for file (Kein Platz im Verzeichnis für die edierte Datei)
[EDLIN]
Sie haben versucht, eine Datei im Basisverzeichnis zu erzeugen oder dort zu speichern, aber entweder ist es voll oder Sie haben ein ungültiges Laufwerk oder einen ungültigen Dateinamen angegeben. Überprüfen Sie die Befehlszeile für den Aufruf von EDLIN. Ist sie in Ordnung, so rufen Sie CHKDSK für die Disk im betreffenden Laufwerk auf. Wenn der Statusreport ein volles Verzeichnis ausweist, andererseits aber noch ausreichend Speicherplatz auf der Disk zur Verfügung ist, sollten Sie die Datei in einem Unterverzeichnis anlegen, da Unterverzeichnisse in Bezug auf die Anzahl der Dateieinträge im Gegensatz zum Basisverzeichnis nicht beschränkt sind. Sie können auch eine neue Diskette einlegen.

No room in root directory (Kein Platz mehr im Basisverzeichnis)
[LABEL]
Das Basisverzeichnis enthält keinen Platz mehr für eine Diskkennung. Löschen Sie eine Datei oder verlagern Sie sie in ein Unterverzeichnis, um Platz im Basisverzeichnis zu schaffen.

No sub-directories exist (Keine Unterverzeichnisse)
[TREE]
Sie haben den TREE–Befehl mit dem Schalter /S aufgerufen. Das Verzeichnis enthält jedoch keine Unterverzeichnisse.

No such file or directory (Keine solche Datei oder kein solches Verzeichnis)
[BACKUP][FC][RESTORE]
Eine oder mehrere von Ihnen angebene Dateien oder Verzeichnisse existieren nicht.

***** Not able to backup/restore file ***** (*** Datei kann nicht gesichert/zurückgeschrieben werden ***)
[BACKUP]
Diese Meldung kann angezeigt werden, wenn die Quelldatei oder die Zieldisk einen Fehler enthält. Versuchen Sie das Problem mit dem CHKDSK–Befehl für die Quelldisk zu lösen.

Not a graphics printer file (Keine Grafik–Druckdatei)
[GRAPHICS]
Die zu druckende Datei enthält keine grafischen Darstellungen.

Not enough memory (Speicher nicht ausreichend)
[JOIN][SHARE][SUBST]
Der angegebene Befehl kann wegen Speicherplatzmangel nicht ausgeführt werden.

Not enough room to merge the entire file (Speicher nicht ausreichend, um die gesamte Datei zu laden)
[EDLIN]
Sie haben versucht, mit dem T (Lade)–Befehl eine Datei zur edierten Datei in den Speicher hinzuzuladen, und der Speicherplatz reicht nicht aus. Sie müssen durch Wegschreiben oder Löschen eines Teils der edierten Datei Platz schaffen.

Not found (Nicht gefunden)
[EDLIN]
Sie haben einen S (Suche)– oder R (Ersetze)–Befehl gegeben, und die betreffende Zeichenkette existiert im restlichen Teil der Datei nicht.

Not ready error reading/writing drive *x:* (Nicht–Bereit Fehler beim Lesen/Schreiben auf Laufwerk *x:*)
[MS–DOS]
Dies ist eine Geräte–Fehlermeldung (s. Kapitel A.1).

O.K.?
[EDLIN]
Diese Anfrage wird während der Ausführung eines S (Suche)– oder R (Ersetze)–Befehls von EDLIN angezeigt. Wird eine beliebige Taste außer der Y–Taste (Yes=Ja) oder der Eingabetaste gedrückt, wird weiter gesucht oder ersetzt.

Out of environment space (Umfeld–Zeichentabelle voll)
[COMMAND][MS–DOS]
Die Umfeld–Zeichentabelle kann keine weiteren Definitionen mehr aufnehmen. Vergrößern Sie die Tabelle entweder mit dem /E–Schalter beim COMMAND–Befehl oder löschen Sie mit dem SET–Befehl nicht mehr benötigte Eintragungen. Beide Befehle sind im Kapitel 6.3.1 ausführlich beschrieben.

Parameters not compatible (Parameter nicht kompatibel)
[FORMAT][REPLACE]
Sie haben zwei Schalter benutzt, die zusammen nicht erlaubt sind.

Parameters not compatible with fixed disk (Parameter bei Festplatte nicht kompatibel)
[FORMAT]
Sie haben einen Schalter angegeben, der mit dem bezogenen Festplattenlaufwerk nicht kompatibel ist.

Parameters not supported (Parameter nicht unterstützt)
[FORMAT][MS–DOS]
Sie haben ungültige Parameter beim Befehlsaufruf angegeben. Überprüfen Sie Ihre Eingabe.

Parameters not supported by drive (Parameter vom Laufwerk nicht unterstützt)
[FORMAT]

Der FORMAT–Befehl zeigt diese Meldung, wenn der Gerätetreiber für das angegebene Laufwerk keine ursprünglichen IOCTL–Funktionsaufrufe unterstützt.

Path(name) too long (Pfad(name) zu lang)
[PRINT][REPLACE][XCOPY]
Der angegebene Pfadname ist länger als 63 Zeichen. Sie müssen das Verzeichnis ggf. wechseln, um den Befehl auf tief geschachtelte Unterverzeichnisse anzuwenden.

Path not found (Suchpfad nicht gefunden)
[CHKDSK][REPLACE][SUBST][XCOPY]
Bei dem Befehl haben Sie einen ungültigen Suchpfad angegeben.

Press any key to begin adding (replacing) file(s) (Beliebige Taste zum Hinzufügen oder Ersetzen von Dateien drücken)
[REPLACE]
Wenn Sie den /W–Schalter angeben, wartet REPLACE mit dem Start, bis Sie eine Taste drücken.

Press any key to begin formatting *x:* (Beliebige Taste zum Formatieren auf Laufwerk *x:* drücken)
[FORMAT]
Diese Meldung wird angezeigt, wenn Sie eine Festplatte formatieren wollen. Drücken Sie eine beliebige Taste oder Ctrl–Break, wenn Sie abbrechen wollen.

Press any key to begin recovery of the *n* file(s) on drive *x:* (Beliebige Taste zum Wiederherstellen der *n* Datei(en) auf Laufwerk *x*: drücken)
[RECOVER]
Diese Meldung wird vor dem Beginn der Wiederherstellung von Teilen einer Disk angezeigt. Wiederhergestellte Dateien erhalten die Namen FILE*nnnn*-.REC, wobei *nnnn* eine fortlaufende Nummer, beginnend mit 0001, ist. Soll nichts wiederhergestellt werden, so kann mit Ctrl–Break abgebrochen werden.

Press any key when ready . . . (Wenn bereit, beliebige Taste drücken)
[DISKCOMP][DISKCOPY]
Diese Anzeige gibt Ihnen die Möglichkeit, vor dem Beginn des Vergleiches oder der Duplizierung die geeigneten Disketten einzulegen. Drücken Sie dann eine beliebige Taste, oder brechen Sie mit Ctrl–Break ab.

Printer error (Druckerfehler)
[MODE]
Der Drucker ist ausgeschaltet oder nicht bereit.

Printer lines per inch set (Druckzeilen/Zoll gesetzt)
[MODE]
MODE meldet das erfolgreiche Setzen des Zeilenabstandes auf dem Drukker.

PRINT queue is empty (PRINT–Warteschlange ist leer)
[PRINT]
Die PRINT–Warteschlange enthält keine Dateien, die auf den Ausdruck warten.

PRINT queue is full (PRINT–Warteschlange ist voll)
[PRINT]
Die PRINT–Warteschlange enthält bereits 10 auf den Ausdruck wartende Dateien. Mit dem /Q–Schalter können Sie beim Aufruf von PRINT die Warteschlange bis auf 32 Einträge erweitern.

Probable non–DOS disk
Continue (Y/N)? (Wahrscheinlich keine MS–DOS–Disk. Fortsetzung (J/N)?)
[CHKDSK]
Die verwendete Disk wird von der aktuellen MS–DOS–Version nicht akzeptiert. Entweder wurde die Disk auf einem anderen System in einem Format formatiert, das nicht von der aktuellen MS–DOS–Version unterstützt wird oder es ist keine MS–DOS–Disk. Fahren Sie nicht mit der Überprüfung fort, wenn CHKDSK diese Meldung bei einer Diskette anzeigt. Wird sie bei einer Festplatte angezeigt, so ist die MS–DOS-Kenninformation zerstört. In diesem Fall können Sie CHKDSK durch Eingabe von Y (Yes = Ja) fortsetzen. Die Dateizuordnungstabelle ist zerstört und die Disk damit nicht mehr brauchbar.

Processing cannot continue (Verarbeitung kann nicht fortgesetzt werden)
[CHKDSK]
Ihr Computer verfügt nicht über ausreichend Speicherplatz, um diese Disk mit CHKDSK zu überprüfen.

Program too big to fit in memory (Programm für Speicher zu groß)
[MS–DOS]

Für das aufgerufenen Anwenderprogramm ist der Speicher nicht ausreichend. Möglicherweise sind einige bereits ausgeführte Programme noch speicherresident. Wenn Sie nach einem Neustart von MS–DOS diese Meldung wieder erhalten, müssen Sie Ihren Computer mit mehr Hauptspeicherplatz ausrüsten.

Read error in *Dateiname* (Lesefehler in *Dateiname*)
[EDLIN][FIND]
MS–DOS kann die Datei wegen eines Lesefehlers nicht vollständig einlesen.

Read fault error reading drive *x:* (Lesefehler auf Laufwerk *x:*)
[MS–DOS]
Dies ist eine Gerätefehlermeldung (s. Kapitel A.1).

Reading source file(s)... (Quelldatei(en) wird/werden gelesen)
[XCOPY]
Mit dieser informativen Meldung zeigt XCOPY an, daß es begonnen hat, die angegebene(n) Quelldatei(en) zu lesen.

Reinsert diskette for drive *x:* (Diskette in Laufwerk *x:* wieder einlegen)
[FORMAT]
Legen Sie die formatierte Diskette wieder in das angegebene Laufwerk ein.

Replace *Dateiname*? (Y/N) (*Dateiname* ersetzen? (J/N))
[REPLACE]
Sie haben REPLACE mit dem /W–Schalter aufgerufen und werden jetzt aufgefordert zu bestätigen, ob die angegebene Datei ersetzt werden soll (Antwort Y (Yes = Ja) oder N (No = Nein)).

Replacing *Dateiname* (*Dateiname* wird ersetzt)
[REPLACE]
REPLACE informiert Sie, daß die angegebene Datei, die auf Ihrer Disk existiert, ersetzt wird.

Requested Screen Shift out of range (Angegebene Bildschirmanzeige nicht weiter verschiebbar)
[MODE]
Die Bildschirmanzeige kann nicht weiter nach rechts oder links verschoben werden.

Resident part of PRINT installed (Residenter Teil von PRINT geladen)
[PRINT]
Sobald Sie PRINT das erste Mal aufrufen, erhalten Sie diese Meldung. Damit PRINT im Hintergrund von anderen Programmen ablaufen kann, muß ein Teil des PRINT–Programms (einige tausend Bytes) resident im Speicher installiert werden.

Resident portion of MODE loaded (Residenter Teil von MODE geladen)
[MODE]
Sobald Sie MODE das erste mal aufrufen, erhalten Sie diese Meldung. Bestimmte Funktionen von MODE (z.B. Verschieben der Bildschirmanzeige) können dann ohne neues Laden ausgeführt werden.

Restore file sequence error (Reihenfolgefehler beim Zurückspeichern)
[RESTORE]
Sie haben versucht, gesicherte Dateien in der falschen Reihenfolge zurückzuspeichern. Die Sicherungsdisketten müssen in derselben Reihenfolge eingelegt werden wie beim Sicherungslauf mit BACKUP.

***** Restoring files from drive *x:* *****
Diskette *n* (*** Dateien werden von Laufwerk *x:* zurückgeschrieben. Diskette *n*)
[RESTORE]
Während des Rückspeicherungsprozesses zeigt RESTORE diese informative Meldung an.

Resynch failed. Files are too different(Keine Übereinstimmung. Dateien sind zu verschieden)
[FC]
FC vergleicht den Anteil zweier Dateien, der in den Hauptspeicher paßt. Wenn keine Zeile dieses Teils mit der verglichenen Datei übereinstimmt, wird diese Meldung angezeigt.

SECOND Diskette bad or incompatible (Zweite Diskette fehlerhaft oder inkompatibel)
[DISKCOMP]
Die zweite Diskette hat nicht dasselbe Format wie die erste oder DISKCOMP kann das Format der zweiten Diskette nicht erkennen. Starten Sie CHKDSK für die zweite Diskette, um das Problem zu identifizieren.

Sector not found error reading/writing drive *x:* (Sektor beim Lesen/Schreiben auf Laufwerk *x:* nicht gefunden)
[MS–DOS]
Dies ist eine Gerätefehlermeldung (s. Kapitel A.1).

Sector size too large in file *Dateiname* (Sektorgröße in Datei *Dateiname* zu groß)
[MS–DOS]
Der angegebene Gerätetreiber in der CONFIG.SYS–Datei benutzt eine Sektorgröße, die größer ist, als die bei allen anderen Treibern spezifizierte. Sie können diesen Gerätetreiber nicht benutzen.

Seek error reading/writing drive *x:* (Positionierungsfehler beim Lesen/Schreiben auf Laufwerk *x:*)
[MS–DOS]
Dies ist eine Geräte–Fehlermeldung (s. Kapitel A.1).

SHARE already installed (SHARE–Hilfsprogramm bereits installiert)
[SHARE]
Die Unterstützung für gemeinsamen Dateizugriff kann nur einmal geladen werden.

Sharing violation reading drive *x:* (Dateizugriff zum Lesen auf Laufwerk *x:* nicht möglich)
Dies ist eine Geräte–Fehlermeldung (s. Kapitel A.1).

SORT: Incorrect DOS version (SORT: Ungültige MS–DOS–Version)
[SORT]
Der Filterbefehl SORT läuft nur unter MS-DOS-Version 2.0 oder jünger.

SORT: Insufficient disk space (SORT: Disk–Speicherplatz nicht ausreichend)
[SORT]
Die sortierte Datei paßt nicht mehr auf die Disk im angegebenen Laufwerk. Rufen Sie SORT erneut mit einer Ausgabedatei auf einem anderen Laufwerk auf.

SORT: Insufficient memory (SORT: Hauptspeicher nicht ausreichend)
[SORT]
Der Filterbefehl lädt die zu sortierende Datei komplett in den Hauptspeicher.

Diese Meldung bedeutet, daß der freie Speicher zum Ausführen der Sortierung nicht mehr ausreicht.

Source and target drives are the same (Quell– und Ziellaufwerk gleich)
[BACKUP][RESTORE]
Bei BACKUP oder RESTORE wurde für Quelle und Ziel dasselbe Laufwerk angegeben. Überprüfen Sie Ihre Eingabe.

Source disk is Non–removable (Quelldisk ist eine Festplatte)
[BACKUP]
Diese informative Meldung zeigt BACKUP an, wenn die Quelldisk beim Sicherungslauf eine Festplatte ist.

Source disk does not contain backup files (Quelldisk enthält keine Sicherungsdateien)
[RESTORE]
Sie haben versucht, Sicherungsdateien von einer Disk zurückzuspeichern, die keine Sicherungsdateien enthält. Legen Sie die richtige Disk ein und wiederholen Sie den Befehl.

Source is a floppy/hard disk (Quelle ist eine Diskette/Festplatte)
[RESTORE]
Mit dieser informativen Meldung zeigt RESTORE den Typ der Quelldisk an.

Source path required (Suchpfad für Quelle erforderlich)
[REPLACE]
Sie haben beim REPLACE–Befehl keinen Suchpfad für die Quelldatei(en) angegeben. Überprüfen Sie Ihre Eingabe.

Specified drive does not exist, or is non–removable (Angegebenes Laufwerk existiert nicht oder ist eine Festplatte)
[DISKCOMP][DISKCOPY]
DISKCOMP und DISKCOPY sind ausschließlich diskettenbezogene Befehle. Sie können nicht auf Festplatten angewendet werden. Überprüfen Sie Ihre Eingabe.

Specified MS–DOS search directory bad (Angegebenes MS–DOS–Suchverzeichnis ist fehlerhaft)
[MS–DOS]
Die CONFIG.SYS–Datei enthält einen fehlerhaften SHELL–Befehl (s.a. Kapitel 2.3). Stellen Sie sicher, daß die COMMAND.COM–Datei existiert und das MS–DOS sie finden kann.

Strike a key when ready. . . (Wenn bereit, eine Taste drücken)
[MS–DOS]
Diese Meldung erscheint während der Bearbeitung eines MS–DOS–Befehls immer zusammen mit einer anderen Meldung. Sie erscheint außerdem, wenn Sie den Stapelbefehl PAUSE verwendet haben. Im allgemeinen fordert MS–DOS zum Diskettenwechsel auf, und fährt mit der Arbeit fort, wenn Sie eine Taste betätigt haben. Sie können alle Zeichentasten, die Leertaste oder die Eingabetaste betätigen.

Syntax error (Syntaxfehler)
[ATTRIB][FIND][MS–DOS]
Sie haben einen Befehl fehlerhaft eingegeben. Überprüfen Sie Ihre Eingabe. Vergessen Sie nicht, die Such–Zeichenkette beim FIND–Befehl in Anführungszeichen (") einzukleiden.

System transferred (System übertragen)
[FORMAT][SYS]
Diese informative Meldung wird angezeigt, wenn die Systemdateien nach einem FORMAT /S– oder einem SYS–Befehl auf die Disk im bezogenen Laufwerk übertragen wurden.

Target cannot be used for backup (Zieldisk kann nicht zur Sicherung benutzt werden)
[BACKUP]
Die Zieldisk hat entweder ein inkompatibles Format oder ist fehlerhaft. Verwenden Sie die Disk nicht, formatieren Sie sie vorher oder rufen Sie CHKDSK auf, um das Problem zu diagnostizieren.

Target disk is Non–removable (Zieldisk ist eine Festplatte)
[BACKUP]
Diese informative Meldung zeigt BACKUP an, wenn die Zieldisk beim Sicherungslauf eine Festplatte ist.

Target diskette is write protected (Zieldiskette ist schreibgeschützt)
[DISKCOPY]
Die Zieldiskette trägt entweder eine Schreibschutzlasche über der Schreibschutzkerbe oder hat keine Schreibschutzkerbe. Im ersteren Fall sollten Sie gut überlegen, ob Sie Daten auf der Disk überschreiben wollen. Falls ja, entfernen Sie die Schreibschutzlasche und wiederholen Sie den DISKCOPY–Befehl. Disketten ohne Schreibschutzkerbe können nicht als Zieldiskette für DISKCOPY benutzt werden.

Target diskette may be unusable (Zieldiskette möglicherweise unbrauchbar)
[DISKCOPY]
Entweder kann DISKCOPY das Format der Zieldiskette nicht erkennen oder diese ist fehlerhaft. Versuchen Sie, die Diskette mit FORMAT neu zu formatieren oder rufen Sie CHKDSK auf, um das Problem zu diagnostizieren.

Target is a flopp/hard disk (Zieldisk ist eine Diskette/Festplatte)
[BACKUP]
Mit dieser informativen Meldung zeigt BACKUP den Typ der Zieldisk an.

Target ist full (Zieldisk ist voll)
[RESTORE]
RESTORE vermag keine weiteren Sicherungsdateien zurückspeichern, weil die Zieldisk voll ist. Sie müssen entweder auf dieser Disk nicht mehr benötigte Dateien löschen oder eine andere Disk verwenden.

Target is Non-removable(Zieldisk ist eine Festplatte)
[RESTORE]
Diese informative Meldung zeigt RESTORE an, wenn die Zieldisk beim Zurückspeichern eine Festplatte ist.

Terminate batch job (Y/N)? (Stapeljob abbrechen (J/N)?)
[MS–DOS]
Wenn Sie Ctrl–Break oder Ctrl–C während der Abbarbeitung einer Stapeldatei drücken, erscheint diese Meldung. Mit Y (Yes = Ja) wird die Bearbeitung beendet. Bei N (No = Nein) wird der nächste Befehl der Stapeldatei bearbeitet.

The last file was not restored (Die letzte Datei wurde nicht zurückgespeichert)
[RESTORE]
Die Zieldisk verfügt über nicht mehr ausreichenden Speicherplatz, um die letzte Sicherungsdatei von der Sicherungsdiskette aufzunehmen oder die letzte Datei enthält Fehler. Starten Sie CHKDISK, um das Problem zu diagnostizieren.

Too many files open (Zu viele Dateien geöffnet)
[EDLIN][LABEL]
EDLIN kann wegen zu wenigen Dateipuffern die .BAK–Datei nicht eröffnen oder LABEL kann aus dem gleichen Grund keine Diskkennung schreiben. Erhöhen Sie die beim FILES–Befehl in der CONFIG.SYS–Datei (s. Kapitel 2.3) anzugebende Zahl von gleichzeitig geöffneten Dateien.

Track 0 bad – disk unusable (Spur 0 fehlerhaft – Disk unbrauchbar)
[FORMAT]
Der FORMAT–Befehl kann zwar defekte Spuren kennzeichnen, nicht jedoch die äußerste Spur (Spur 0). Sie müssen eine andere Disk verwenden.

Unable to create directory (Verzeichnis kann nicht eingerichtet werden)
[MKDIR][XCOPY]
MS–DOS kann das angegebene Verzeichnis nicht einrichten. Stellen Sie sicher, daß kein Namenskonflikt vorliegt. Es kann eine Datei desselben Namens existieren oder die Disk ist voll.

Unable to erase (Löschen nicht möglich)
[BACKUP]
Die Dateien auf der Zieldiskette können nicht gelöscht werden. Stellen Sie sicher, daß weder die Dateien auf der Zieldisk noch die Zieldisk selbst schreibgeschützt sind.

Unable to shift Screen (Bildschirmanzeige nicht verschiebbar)
[MODE]
Das Testbild zum Einstellen der Bildschirmanzeige ist nicht weiter nach rechts oder links verschiebbar.

Unexpected DOS Error *n* (Unerwarteter MS–DOS–Fehler *n*)
[REPLACE]
Ein unerwarteter MS–DOS–Fehler mit dem Fehlercode *n* ist während des Zurückspeicherns von Sicherungsdateien aufgetreten.

Unrecognized command in CONFIG.SYS (Unbekannter Befehl in CONFIG.SYS–Datei)
[MS–DOS]
Beim Abarbeiten der CONFIG.SYS–Datei wurde ein nicht interpretierbarer Befehl gefunden. In Kapitel 2.3 sind alle gültigen CONFIG.SYS–Befehle aufgeführt.

Unrecognized printer (Unbekannter Drucker)
[GRAPHICS]
Sie haben versucht, einen Drucker unter einem unbekannten Namen zu benutzen. Überprüfen Sie Ihre Eingabe und prüfen Sie, ob Sie einen gültigen Druckernamen benutzt haben (s.a. Kapitel 4.1.6).

Unrecognized printer port (Unbekannter Druckeranschluß)
[GRAPHICS]
Sie haben einen ungültigen Druckergerätenamen angegeben. Möglicherweise müssen Sie mit dem MODE–Befehl den Druckeranschluß noch spezifizieren.

Unrecoverable error in directory
Convert directory to file (Y/N)? (Nicht behebbarer Fehler im Verzeichnis. Verzeichnis in Datei umwandeln (J/N)?)
[CHKDSK]
CHKDSK kann einen Verzeichnisfehler nicht beheben und bietet die Umwandlung in eine Datei an. Wenn Sie mit Y (Yes = Ja) antworten, wird in eine Datei umgewandelt, so daß Sie später das Verzeichnis retten können. Wird mit N (No = Nein) geantwortet, so kann zukünftig das Verzeichnis und ggf. dort eingerichtete Unterverzeichnisse nicht mehr benutzt werden.

Unrecoverable read/write error on drive *x:* (Nicht behebbarer Lese–/Schreibfehler auf Laufwerk *x:*)
[MS–DOS]
Dies ist eine Gerätefehlermeldung (s. Kapitel A.1).

Usage: fc [/a] [/b] [/c] [/l] [/lb n] [/w]
[t] [/n] [/nnnn] file1 file 2 (Gültig sind: fc. . . .)
[FC]
Sie haben einen ungültigen Schalter bei FC benutzt. Überprüfen Sie Ihre Eingabe anhand der angezeigten, erlaubten Parameter.

VERIFY is off/on (VERIFY ist EIN/AUS)
[MS–DOS]
Diese informative Meldung zeigt an, ob der Schalter, der die Lesen-nach-Schreiben–Prüfung bei Disk–Ausgabe steuert, ein– oder ausgeschaltet ist, wenn der VERIFY–Befehl ohne Parameter eingegeben wird.

Volume in drive *x:* **has no label** (Disk in Laufwerk *x:* hat keine Kennung)
[DIR][LABEL][VOL]
Diese informative Meldung wird angezeigt, wenn beim DIR–, LABEL– oder VOL–Befehl keine Kennung bei der Disk im angegebenen Laufwerk gefunden wird.

Volume in drive *x:* is *Name* (Disk in Laufwerk *x:* hat die Kennung *Name*)
[DIR][LABEL][VOL]
Diese informative Meldung wird angezeigt, wenn beim DIR–, LABEL– oder VOL–Befehl eine Kennung bei der Disk im angegebenen Laufwerk gefunden wird.

Volume label (11 characters, ENTER for none)? (Diskkennung (11 Zeichen, Eingabetaste für keine Kennung)?)
[FORMAT][LABEL]
Mit dieser Anzeige fordert der FORMAT /V– oder der LABEL–Befehl zur Eingabe einer bis zu 11 Zeichen langen Diskkennung auf. Wollen Sie keine Diskkennung vergeben, brauchen Sie nur die Eingabetaste zu drücken. Es ist jedoch sinnvoll, besonders Disketten Kennungen zu geben, um schnell einen Überblick über den Inhalt zu bekommen.

WARNING, ALL DATA ON NON-REMOVABLE DISK
DRIVE *x:* WILL BE LOST!
Proceed with Format (Y/N)? (ACHTUNG: ALLE DATEN AUF DER FESTPLATTE IN LAUFWERK *x:* WERDEN GELÖSCHT! Formatieren fortsetzen (J/N)?)
[FORMAT]
Diese Warnmeldung mit angeforderter Bestätigung erhalten Sie, wenn Sie eine Festplatte formatieren, die bereits Daten enthält. Wenn Sie mit Y (Yes = Ja) antworten, werden alle Daten gelöscht. Wenn Sie die Daten retten wollen, antworten Sie mit N (No = Nein). Sie können dann z.B. erst alle Daten auf Disketten kopieren (BACKUP) und dann die Formatierung vornehmen.

Warning – directory full (ACHTUNG: Verzeichnis voll)
[RECOVER]
Während des Wiederherstellungsprozesses mit RECOVER ist das Basisverzeichnis voll geworden. Löschen Sie nicht mehr benötigte Dateien aus dem Basisverzeichnis und wiederholen Sie den Befehl.

Warning! Diskette is out of sequence
Replace diskette or continue if okay
Strike any key when ready (ACHTUNG: Diskette nicht folgerichtig. Diskette wechseln oder fortfahren. Wenn bereit, eine Taste drücken)
[RESTORE]
Sie haben beim Zurückspeichern von Sicherungsdateien die Reihenfolge der Sicherungsdisketten durcheinander gebracht. Halten Sie dieselbe Reihenfolge ein, in der die Disketten mit BACKUP erstellt wurden.

Warning! File *Dateiname*
is a hidden/read-only file
Replace the file (Y/N)? (ACHTUNG: Datei *Dateiname* ist eine verborgene/schreibgeschützte Datei. Überschreiben (J/N)?)
[RESTORE]
Beim Zurückspeichern von Sicherungsdateien wird diese Meldung angezeigt, wenn eine verborgene oder schreibgeschützte Datei vorkommt. Wird mit Y (Yes = Ja) geantwortet, so wird überschrieben; bei N (No = Nein) wird diese Datei übergangen.

Warning! File *Dateiname*
was changed after it was backed up
Replace the file (Y/N)? (ACHTUNG: Datei *Dateiname* wurde nach der Sicherung verändert. Überschreiben (J/N)?)
[RESTORE]
Es soll eine Datei zurückgespeichert werden, die auf der Zieldisk existiert und nach der Sicherung verändert wurde. Wird mit Y (Yes = Ja) geantwortet, so wird überschrieben; bei N (No = Nein) wird diese Datei übergangen.

Warning! Files in the target
drive \BACKUP/root directory will be erased (ACHTUNG: Dateien im \BACKUP–/Basis–Verzeichnis der Zieldisk werden überschrieben)
[BACKUP]
BACKUP findet auf der Disk im Ziellaufwerk in dem \BACKUP– oder Basis–Verzeichnis Dateien und Sie haben nicht den /A–Schalter für Anfügen angegeben.

Warning! No files were found to back up (ACHTUNG: keine Dateien zum Sichern gefunden)
[BACKUP]
Auf der angegebenen Disk konnte BACKUP keine zu sichernden Dateien finden.

Warning: No files were found to restore(ACHTUNG: keine Dateien zum Zurückspeichern gefunden)
[RESTORE]
Die von Ihnen angegebenen zurückzuspeichernden Dateien sind nicht auf der angegebenen Disk enthalten.

Warning: Read error in EXE file (ACHTUNG: Lesefehler in .EXE–Datei)
[EXE2BIN]
EXE2BIN zeigt an, daß weniger Bytes gelesen wurden als im Programmkopf angegeben. Diese Meldung ist nur als Warnung zu interpretieren.

Write fault error writing drive *x:* (Schreibfehler auf Laufwerk *x:*)
Dies ist eine Gerätefehlermeldung (s. Kapitel A.1).

Write protect error writing drive *x:* (Schreibschutz auf Laufwerk *x:*)
Dies ist eine Gerätefehlermeldung (s. Kapitel A.1)

Wenn Sie Ihre Microsoft Produkt in der BUNDESREPUBLIK DEUTSCHLAND, in der SCHWEIZ, in ÖSTERREICH oder in LIECHTENSTEIN gekauft haben, trifft der folgende Lizenzvertrag für Sie zu:

Microsoft Software Lizenzvertrag

MICROSOFT SOFTWARE LIZENZVERTRAAG

Nachfolgend sind die Vertragsbedingungen für die Benutzung von Microsoft Software durch Sie, den Endverbraucher (im folgenden auch : "Lizenznehmer"), aufgeführt.

Durch Öffnen der versiegelten Diskettenpackung wie auch durch Unterzeichnung der Registrierungskarte erklären Sie sich mit diesen Vertragsbedingungen einverstanden.

Daher lesen Sie bitte den nachfolgenden Text vollständig und genau durch.

Wenn Sie mit diesen Vertragsbestimmungen nicht einverstanden sind, so dürfen Sie die Diskettenpackung nicht öffnen.

Geben Sie in diesem Fall die ungeöffnete Diskettenpackung und alle anderen Teile des erworbenen Produktes (einschl. allen schriftlichen Materials, mitgelieferter Hardware und der Verpackung) unverzüglich dort, wo Sie das Produkt erworben haben, zurück; der Erwerbspreis wird Ihnen voll zurückerstattet.

VERTRAGSBEDINGUNGEN

1. Gegenstand des Vertrages

Gegenstand des Vertrages ist das auf dem Datenträger (Diskette) aufgezeichnete Computerprogramm, die Programmbeschreibung und Bedienungsanleitung, sowie sonstiges zugehöriges schriftliches Material. Sie werden im folgenden auch als "Software" bezeichnet.

Microsoft macht darauf aufmerksam, daß es nach dem Stand der Technik nicht möglich ist, Computer-Software so zu erstellen, daß sie in allen Anwendungen und Kombinationen fehlerfrei arbeitet.

Gegenstand des Vertrages ist daher nur eine Software,die im Sinne der Programmbeschreibung und der Benutzungsanleitung grundsätzlich brauchbar ist.

2. Umfang der Benutzung

Microsoft gewärt Ihnen für die Dauer dieses Vertrages das einfache nicht-ausschließliche und persönliche Recht (im folgenden auch als "Lizenz" bezeichnet), die beiliegende Kopie der Microsoft-Software auf einem einzelnen Computer (d.h. mit nur einer einzigen Zentraleinheit (CPU), und nur an einem Ort zu benutzen. Ist dieser einzelne Computer ein Mehrbenutzersystem, so gilt dieses Benutzungsrecht für alle Benutzer dieses einen Systems.

Als Lizenznehmer dürfen Sie Software in körperlicher Form (d.h. auf einem Datenträger abgespeichert) von einem Computer auf einen anderen Computer übertragen, vorausgesetzt, daß sie zu irgendeinem Zeitpunkt auf immer nur einem einzelnen Computer genutzt wird.

Eine weitergehende Nutzung ist nicht zulässig.

3. Besondere Beschränkungen

Dem Lizenznehmer ist untersagt

a) ohne vorherige schriftliche Einwilligung von Microsoft die Software oder das zugehörige schriftliche Material an einen Dritten zu übergeben oder einem Dritten sonstwie zugänglich zu machen,

b) die Software von einem Computer über ein Netz oder einen Datenübertragungskanal auf einen anderen Computer zu übertragen,

c) ohne vorherige schriftliche Einwilligung von Microsoft die Software abzuändern, zu übersetzen, zurückzuentwickeln, zu entkompilieren oder zu entassemblieren,

von der Software abgeleitete Werke zu erstellen oder das schriftliche Material zu vervielfältigen,

es zu übersetzen oder abzuändern oder vom schriftlichen Material abgeleitete Werke zu erstellen.

4. Inhaberschaft an Rechten

Sie erhalten mit dem Erwerb des Produktes nur Eigentum an dem körperlichen Datenträger, auf dem die Software aufgezeichnet ist. Ein Erwerb von Rechten an der Software selbst ist damit nicht verbunden. Microsoft behält sich insbesondere alle Veröffentlichungs-,Vervielfältigungs-, Bearbeitungs- und Verwertungsrechte an der Software vor.

5. Vervielfältigung

Die Software und das zugehörige Schriftmaterial sind urheberrechtlich geschützt.

Soweit die Software nicht mit einem Kopierschutz versehen ist, ist Ihnen das Anfertigen einer einzigen Reservekopie nur zu Sicherungszwecken erlaubt. Sie sind verpflichtet, auf der Reservekopie den Urheberrechtsvermerk der Microsoft anzubringen bzw. ihn darin aufzunehmen. Ein in der Software vorhandener Urheberrechtsvermerk sowie in ihr aufgenommene Registrierungsnummern dürfen nicht entfernt werden.

Es ist ausdrücklich verboten, die Software wie auch das schriftliche Material ganz oder teilweise in ursprünglicher oder abgeänderter Form oder in mit anderer Software zusammengemischter oder in anderer Software eingeschlossener Form zu kopieren oder anders zu vervielfältigen.

6. Übertragung des Benutzungsrechtes

Das Recht zur Benutzung der Software kann nur mit vorheriger schriftlicher Einwilligung von Microsoft und nur unter den Bedingungen dieses Vertrages an einen Dritten übertragen werden.

Verschenken, Vermietung und Verleih der Software sind ausdrücklich untersagt.

7. Dauer des Vertrages

Der Vertrag läuft auf unbestimmte Zeit.

Das Recht des Lizenznehmers zur Benutzung der Software erlischt automatisch ohne Kündigung, wenn er eine Bedingung dieses Vertrages verletzt. Bei Beendigung des Nutzungsrechtes ist er verpflichtet, die Originaldiskette wie alle Kopien der Software einschl. etwaiger abgeänderter Exemplare, sowie das schriftliche Material zu vernichten.

8. Schadensersatz bei Vertragsverletzung

Microsoft macht darauf aufmerksam, daß Sie für alle Schäden aufgrund von Urheberrechtsverletzungen haften, die Microsoft aus einer Verletzung dieser Vertragsbestimmungen durch Sie entstehen.

9. Änderungen und Aktualisierungen

Microsoft ist berechtigt, Aktualisierungen der Software nach eigenem Ermessen zu erstellen.

Microsoft ist nicht verpflichtet, Aktualisierungen des Programms solchen Lizenznehmern zur Verfügung zu stellen, die die Registrierungskarte nicht unterzeichnet an Microsoft zurückgesandt oder die Aktualisierungsgebühr nicht bezahlt haben.

10. Gewährleistung und Haftung der Microsoft

a) Microsoft gewährleistet gegenüber dem ursprünglichen Lizenznehmer, daß zum Zeitpunkt der Übergabe des Datenträgers (Diskette), auf dem die Software aufgezeichnet ist, und die mit der Software zusammen ausgelieferte Hardware unter normalen Betriebsbedingungen und bei normaler Instandhaltung in Materialausführung fehlerfrei sind.

b) Sollte der Datenträger (die Diskette) oder die damit ausgelieferte Hardware fehlerhaft sein, so kann der Erwerber Ersatzlieferung während der Gewährleistungszeit von 6 Monaten ab Lieferung verlangen . Er muß dazu die Diskette, die eventuell mit ihr ausgelieferte Hardware einschl. der Reservekopie und dem schriftlichen Material, und einer Kopie der Rechnung/Quittung an Microsoft oder an den Händler, von dem das Produkt bezogen wurde, zurückgeben.

c) Wird ein Fehler im Sinne von Zif. 10b nicht innerhalb angemessener Frist durch eine Ersatzlieferung behoben, so kann der Erwerber nach seiner Wahl Herabsetzung des Erwerbspreises oder Rückgängigmachen des Vertrages verlangen.

d) Aus den vorstehend unter 1. genannten Gründen übernimmt Microsoft keine Haftung für die Fehlerfreiheit der Software. Insbesondere übernimmt Microsoft keine Gewähr dafür, daß die Software den Anforderungen und Zwecken des Erwerbers genügt oder mit anderen von ihm ausgewählten Programmen zusammenarbeitet. Die Verantwortung für die richtige Auswahl und die Folgen der Benutzung der Software sowie der damit beabsichtigten oder erzielten Ergebnisse trägt der Erwerber .

Das gleiche gilt für das die Software begleitende schriftliche Material.

Ist die Software nicht im Sinne von 1. grundsätzlich brauchbar, so hat der Erwerber das Recht, den Vertrag rückgängig zu machen . Das gleiche Recht hat Microsoft, wenn die Herstellung von im Sinne von 1. brauchbarer Software mit angemessenem Aufwand nicht möglich ist.

e) Microsoft haftet nicht für Schäden, es sei denn, daß ein Schaden durch Vorsatz oder grobe Fahrlässigkeit seitens Microsoft verursacht worden ist. Gegenüber Kaufleuten wird auch die Haftung für grobe Fahrlässigkeit ausgeschlossen.

Eine Haftung wegen evtl. von Microsoft zugesicherten Eigenschaften bleibt unberührt. Eine Haftung für Mangelfolgeschäden, die nicht von der Zusicherung umfaßt sind, ist ausgeschlossen.

11. Ist der Lizenznehmer Vollkaufmann, so wird auf diesen Vertrag das Recht des Staates Washington (USA) angewendet. In diesem Fall ist weiter die Zuständigkeit der im Staate Washington gelegenen Staatsgerichte und Bundesgerichte vereinbart.

Falls Sie Fragen zu dem Microsoft-Software-Lizenzvertrag haben oder Microsoft ansprechen wollen, wenden Sie sich bitte schriftlich an: Microsoft GmbH, Erdinger Landstraße 2, 8011 Aschheim/Dornach.

BESCHEINIGUNG DES HERSTELLERS

Hiermit wird bestätigt, daß der Personal-Computer

COMMODORE PC 10/20/40/A 2000

in Übereinstimmung mit den Bestimmungen der

Amtsblattverfügung Nr. 1046/1984

funk-entstört ist.

Der Deutschen Bundespost wurde das Inverkehrbringen dieses Gerätes angezeigt und die Berechtigung zur Überprüfung eingeräumt.

COMMODORE BÜROMASCHINEN GMBH

CERTIFICATE OF THE MANUFACTURER

Herewith we certify that our device Personal-Computer

COMMODORE PC 10/20/40/A 2000

corresponds to the regulations

Amtsblattverfügung Nr. 1046/1984

is eliminated of radio interference.

The German Bundespost has been informed that this unit is on the market and has got the right to check on the mass production if the limits are kept.

COMMODORE BUSINESS MACHINES LIMITED